Application of H_∞ Optimization Techniques in Image & Signal Processing

by

Jibril Odogba

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

SYSTEM ENGINEERING

June, 1995

Application Of H_∞ Optimization Techniques In Image & Signal Processing

BY

Jibril Odogba

A Thesis Presented to the

FACULTY OF THE COLLEGE OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

Systems Engineering

June 1995

KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

DHAHRAN, SAUDI ARABIA

This thesis, written by

JIBRIL ODOGBA

under the direction of his thesis committee, and approved by all the members, has been presented to and accepted by the Dean, College of Graduate Studies, in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING

Thesis Committee:

Chairman (Dr.Davut Kavranoglu)

Member (Dr. Maamar Bettayeb)

Member(Dr. Lahouari Cheded)

Dr. Khaled S. Al-Sultan
Department Chairman

Dr. Ala H. Rabeh
Dean, College of Graduate Studies

Date: 3/7/95

Dedication

For the Oppressed of the World:

'Cos no Evil by Man against Man

Can go Unpunished,

If not now, then certainly Later,

If not by Man, then surely by ALLAH!

'Cos the Recompense of Evil

Over Good is only Temporal!!

Acknowledgments

Al-hamdillahi! Subhana-lillah! I want to start by thanking GOD Almighty for having spared my life in this temporary land (earth) till now and for all the blesings he has bestowed on me.

My sincerest thanks to my thesis advisor, Dr. Davut Kavranoglu who introduced the exciting subject of H_∞ to me. I have found the many technical sessions which we conducted at both regular and odd hours most valuable.

I have also enjoyed the pleaseant academic and non-academic collaboration of my thesis committee members:-Dr. Maamar Bettayeb and Dr. Lahouari Cheded. Much thanks for your support and worthy contributions.

I am grateful to Prof. M.M. Fahmy who had a direct and indirect influence on this work through the couple of discussions we had, and also for providing a repertoire of images to work with.

Merci boku to my family for their prayers and for keeping the spirit alive. I have also enjoyed a symbiotic relationship with my friends, wellwishers e.t.c.; the list is so long that it is impractical to mention specific names but *you all know who you are.* However, I am impelled to note down my all-time friend Shade Lé Petit Sanni.

Muchas Gracias to you all.

Messrs Shuaib Khan, Asjad Khan and Mutlaq Al-Mutlaq of the CCSE network had to bear with the nightmarish nightly back-up my huge image data base was causing the network and at the same time put up with my ceaseless demands regarding ample memory, et-al. You sure eased a lot for me. Thanks.

I want to thank KFUPM for giving me this chance. May Allah (S.W.T.) continue to shower his limitless bounties on us all.*(Ameen)*.

Contents

List of Figures

List of Tables

Abstract

Name: Jibril Odogba

Title: Application Of H_∞ Optimization Techniques In Image & Signal Processing

Major Field: Systems Engineering

Date of Degree: June 1995

Image processing has been receiving extensive attention in recent years because of its potential applications to a number of fields. Video, Audio, and Images produce vast amount of data which requires a tremendous amount of memory for its storage, and could not possibly be faithfully transmitted due to channel bandwidth limitation. The only solution is to compress the data involved. Six new algorithms are proposed for the data compression problem. Comparisons are made with an implemented Discrete Cosine Transform (DCT)-based compression method. One of the algorithms is modified and applied to a speech signal. The performance and evaluation of these new algorithms are addressed based on fidelity criteria, and implemented examples are given.

Master of Science Degree

King Fahd University of Petroleum and Minerals

Dhahran, Saudi Arabia . June 1995

خلاصة الرسالة

اسم الطالـــب : جبريـــــل اودجبـــا

عنوان الرسالة : تطبيق طرق هـ فى معالجة الصور والاشارات .

التخصـــــص : هندسة النظم .

تاريخ الشهادة : يونيو ١٩٩٥ م

اصبحت معالجة الصور تتلقى قدرا عظيما من الاهتمام لإمكانيه تطبيقها فى عدد من المجالات . إن الفيديو والصوت والصوره ينتج عنهم كم هائل من البيانات . وهذه تحتاج لكميه هائله من خير التخزين ولا يمكن أن تنقل كامله نتيجة القيود على سعه القناه الناقله . ويصبح الحل الوحيد هو ضغط هذه البيانات . وفى هذه الرسالة نقترح سته طرق جديده لضغط البيانات . وقمنا بمقارنه النتائج مع طريقة لضغط المعلومات تعتمد على تحويل جيب التمام المتقطع. وكذلك طورنا احدى الطرق واستخدمناها للاشارات الصوتيه . ويضم البحث تقييم أداء هذه الطرائق المستخدمة حيث كان المعيار هو الجوده وطبق ذلك على أمثله .

درجة الماجستير في العلوم

جامعة الملك فهد للبترول والمعادن

الظهران ، المملكة العربية السعودية

يونيو ١٩٩٥ م

Nomenclature

$\Re$ field of real numbers

$\Re^{n\times m}$ the set of $n \times m$ matrices with real elements

C field of complex numbers

C_+ field of complex right half plane

$C^{n\times m}$ the set of $n \times m$ matrices with complex elements

A^T transpose of a matrix A

A^* complex conjugate transpose of a matrix A

$trace(A)$ the sum of the diagonal elements of a matrix A

$\lambda_i(A)$ the i'th eigenvalue of matrix A

σ_i the i'th Hankel Singular Value

$\rho(A)$ the spectral radius of matrix A

Γ_g the Hankel operator with symbol G

$\|.\|$ norm on $C^{n\times m}$; maximum singular value

$\|G\|_\infty$ $sup_{w\in\Re}\sigma_{max}(G(jw))$

H_∞ Hardy space; space of functions of a complex variable that are nalytic in C_+ and bounded on the jw axis

I_m $m \times m$ identity matrix

O_m $m \times m$ null matrix

SISO Single Input Single Output

MIMO Multiple Input Multiple Output

LS Least Squares

DCT Discrete Cosine Transform

SNR Signal-to-Noise-Ratio

MSE Mean Square Error

SBRI Sorted Balanced Model Reduced Image

NSBRI Non-Sorted Balanced Model Reduced Image

SHRI Sorted Hankel Model Reduced Image

NSHRI Non-Sorted Hankel Model Reduced Image

DCTI Discrete Cosine Transform Image

Chapter 1

Introduction

1.1 General

An image is mathematically represented by a two-dimensional light intensity function $f(x, y)$, where x and y denote spatial co-ordinates and the value of f at any point (x, y) is proportional to the brightness(or gray level) of the image at that point. Figure 1.1 illustrates this convention [1].

Thus a digital image is an image $f(x, y)$ that has been discretized both in spatial co-ordinates and brightness. As such, it can then be represented by a matrix whose row and column indices identify a point in the image and the corresponding matrix element value identifies the gray level at that point. The elements of such a digital array are called image elements, picture elements, pixels or pels, with the last two being commonly-used abbreviations of 'picture elements'.

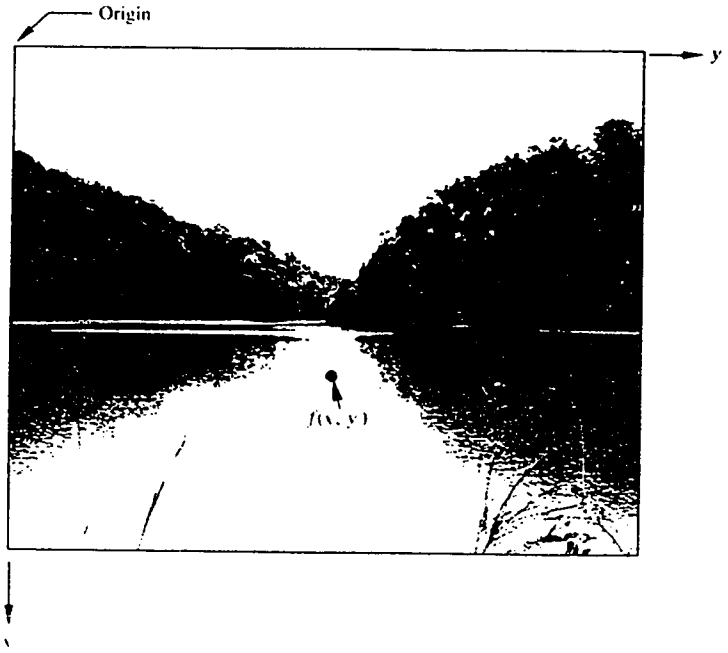


Figure 1.1 : Image representation axis convention.

A two-dimensional (2-D) discrete-space signal or sequence are denoted by a function whose two arguments are integers, while a one-dimensional (1-D) sequence is denoted by a function with only one integer argument. For example, $x(n_1, n_2)$ represents a 2-D sequence whose arguments are defined for all possible integer values of n_1 and n_2 while $x(n)$ represents a 1-D sequence with argument defined for all possible integer values of n.

Examples of a 2-D sequence $x(n_1, n_2)$ and a 1-D sequence $x(n)$ are sketched in Figure 1.2 and Figure 1.3, respectively. In Figure 1.2, the height at (n_1, n_2) represents the amplitude at (n_1, n_2), i.e., $x(n_1, n_2)$.

An important 1-D signal processing application is speech processing [2]. Figure 1.4 shows an example of a speech signal represented both as an analog signal and as a sequence of samples taken at a sampling rate of $8KHz$.

The general problem of image and signal processing is depicted in Figure 1.5 . Image and signal processing involves first obtaining a representation of the image or signal based on a given model and then applying to this representation some higher level transformation in order to put the image or signal into a more convenient form. The last step in the process is the extraction and utilization of the necessary information. This step may be performed either manually by a human being or automatically by machines. By way of example, a system whose function is to automatically identify a speaker from a given set of speakers might use

a time-dependent spectral representation of the speech signal. One possible

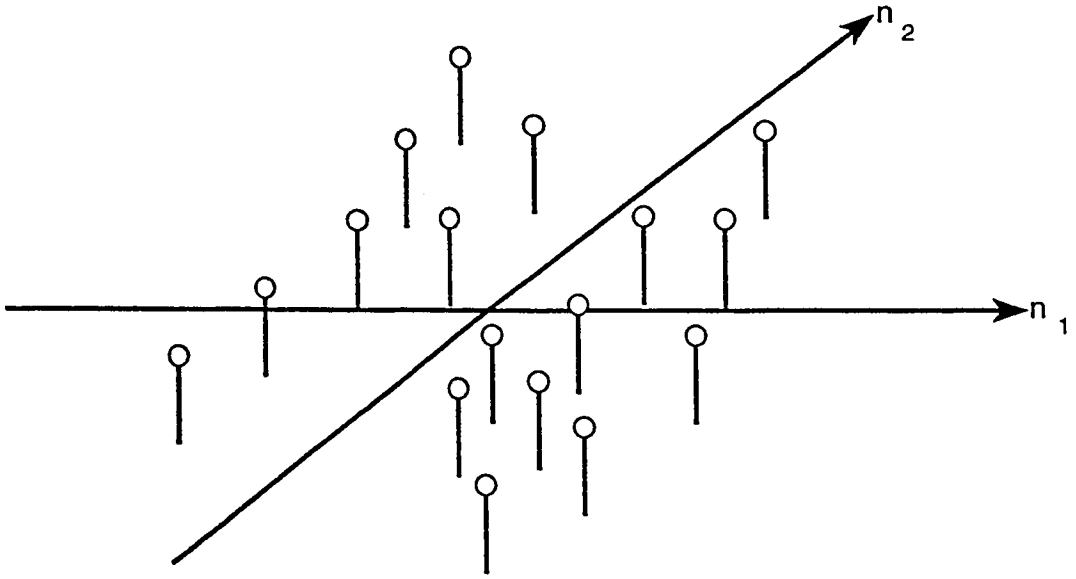


Figure 1.2 : Two-dimensional Sequence

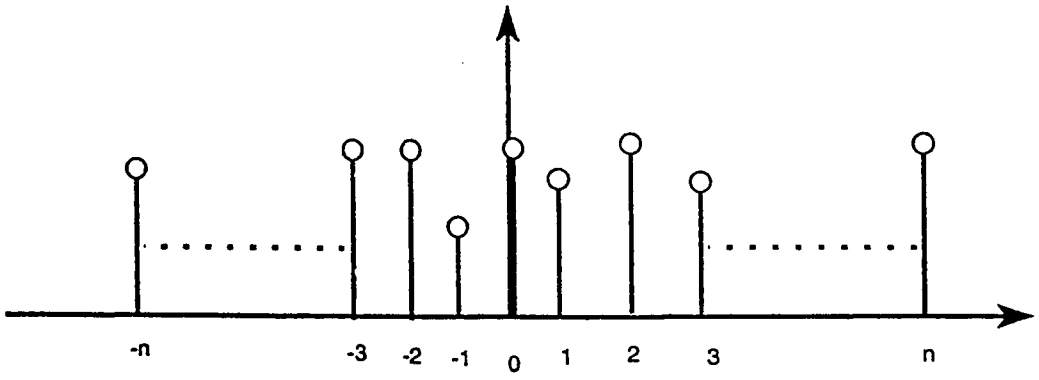


Figure 1.3 : One-dimensional Sequence

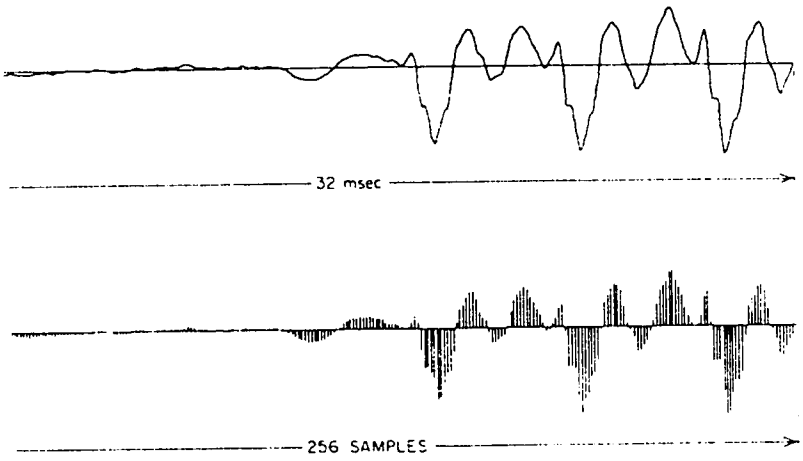


Figure 1.4 : Representation of speech signal.

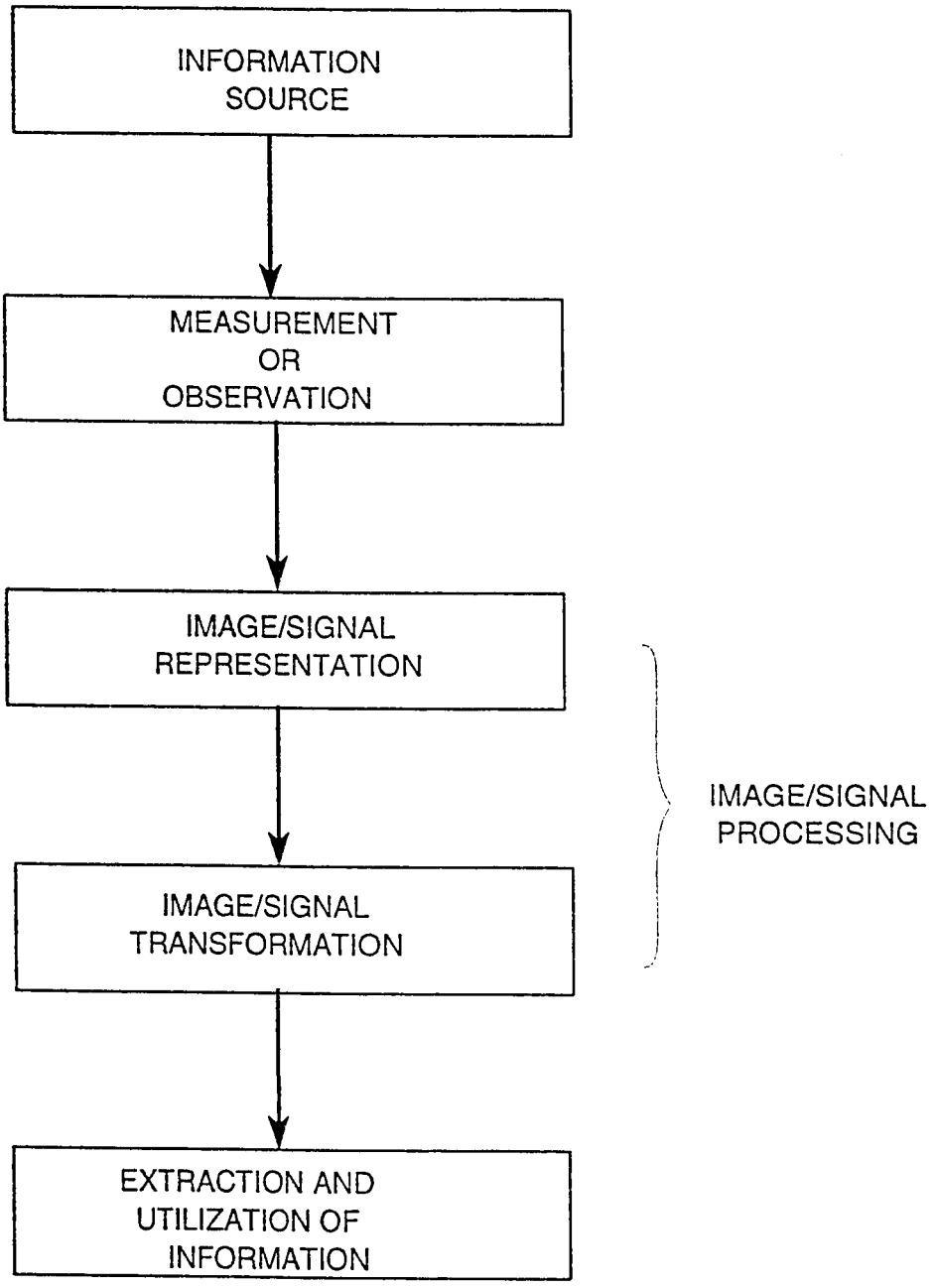


Fig. 1.5 General view of image and signal processing

signal transformation would be to average spectra across an entire sentence, compare the average spectrum to a stored average spectrum template for each possible speaker, and then based on a spectral similarity measurement choose the identity of the speaker. For this example, the "information" is the identity of the speaker.

2-D signal and image processing has been receiving extensive attention in recent years because of its applications to a number of areas. Applications are primarily in transmission and storage of information. Image transmission applications are in broadcast television(tv), remote sensing via satellite, military communications via aircraft, radar and sonar, tele-conferencing, computer communications, facsimile transmission and the like. Image storage is required for example in educational and business documents, legal and criminal records, medical imaging (computer tomography (CT) , magnetic resonance imaging (MRI) and digital radiology), motion pictures, satellite images, weather maps, geological surveys [1, 3, 4, 5].

1.2 Background and Problem Formulation

A vast amount of data is produced by Audio, Images and Video signals. The storage requirements for various media types are illustrated in Table 1.1 [3].

Present multimedia systems require data compression for three reasons [3]:

1. The large storage requirements of multimedia data.

2. Relatively slow storage devices that cannot play multimedia data(specifically

Table 1.1 : Storage requirements for various data types

	Text	Image	Audio	Animation	Video
Object type	ASCII EBCDIC	Bitmapped graphics Still photos Faxes	Noncoded stream of digitized audio or voice	Synched image and stream at 15-19 frames/s	TV analog or digital image with synched streams at 24-30 frames/s
Size and bandwidth	2 KB per page	Sample: 64 KB per image Detailed (color) 7.5 MB per image	Voice/phone 8KHz/ 8 bits (mono) 6-44 KB/s Audio CD DA 44.1 kHz/ 16 bit 176 KB/s	2.5 MB/s for 320x640x16 pixels/frames (16 bit color) 16 frames/s	27.7 MB/s for 640x480x24 pixels per frame (24-bit color) 30 frames/s

KB = Kbytes MB = Mbytes

video) in real time.

3. Channel bandwidth constraint that does not allow for real time video data transmission.

The following motivating examples illustrates the practical necessity for data compression:

- Example 1 : Consider a single frame of a color video with 620 × 560 pixel frames.

  At 24 bits per pixel, this single frame would take up a little over 1Mbyte ($620 \times 560 \times 24 = 1,041,600 bytes = 1.0416 MB$)

  At a real time rate of 30 frames per second, it would require just over 30 Mbytes for one second of video!

  For a storage device to correctly play these color frames in real time, its speed would have to be just over 30 Mbytes/s!!!

  However, today's CD-ROM technology provides a maximum transfer rate of 300Kbytes/s which is hardly any match to the required speed!

- Example 2: Consider converting an analog video signal into a digital one.

  The transmission of a 4MHz T.V signal, sampled at the Nyquist rate with 8 bits per sample.

  Using phase-shift keying (PSK) requires 32MHz, hence an eightfold increase

in bandwidth !!!

- Example 3: Consider the digital coding of human speech to make it suitable for transmission over public telephone channels.

  The frequency spectrum of human speech ranges between 30 and 9,000Hz (for music it can reach 20,000Hz).

  However, most of the content of speech that is important for intelligibility is in the range 300-3,400Hz.

  To avoid aliasing, the signal is bandlimited to 4KHz and then sampled at least at 8KHz as required by Shannon's sampling theorem.

  If we allocate 8bits/sample, the required bit rate would be 64,000 bits per second !!!

  This transmission rate is simply not possible over public telephone lines as the latter have bandwidth limited to between 2.4-4.8Kbits/second.

Hence the amount of data associated with the three examples discussed above is so large that its storage and/or transmission would require large capacity and /or high bit rate. Although the capacities of several storage media (Table 1.2) are substantial, their access speeds are normally inversely proportional to their capacity[4]. Storage and/or transmission of such data would require a large capacity and/or bandwidth, which would be very expensive.

Table 1.2a. Data Volumes of Image Sources(in millions of bytes)

National Archives	12.5×10^9
1 hour of color television	28×10^3
Encyclopeadia Britannica	12.5×10^3
Book(200 pages of text characters)	1.3
One page viewed as an image	0.13

Table 1.2b. Storage Capacities(in millions of bytes)

Human brain	125,000,000
Magnetic cartridge	250,000
Optical disc memory	12,500
Magnetic disc	760
2400-ft magnetic tape	200
Floppy disc	1.25
Solid-state modules	0.25

The only solution is to compress the data. Data compression techniques are concerned with reduction of the number of bits required to store or transmit these data without any appreciable loss of information. Figure 1.6 shows the compressed storage requirement for images and video. This process is called image coding/compression in the image processing literature, speech coding/compression in speech processing, and model reduction/approximation in control and systems theory.

Digital data compression relies on various computational algorithms, implemented either in hardware or software. Application of data compression is also possible in the development of fast algorithms where the number of operations required to implement an algorithm is reduced by working with the compressed data [4]. We can classify compression techniques into lossless and lossy approaches. Lossless techniques can perfectly recover the original data. Lossy techniques recover the original data with some loss of accuracy. The lossy techniques provide higher compression ratios, and are more applied often in image and video compression than lossless techniques [3].

Lossless compression is particulary required in image archiving, for example in the storage of legal, medical records, and business documents where lossy compression is prohibited for legal reasons. Also, it is required in digital radiography where diagnostic accuracy cannot be compromised through loss of information. On the other hand, lossy compression is useful in applications such as broadcast television(TV), videoconferencing, and facsimile transmission, in which a certain amount

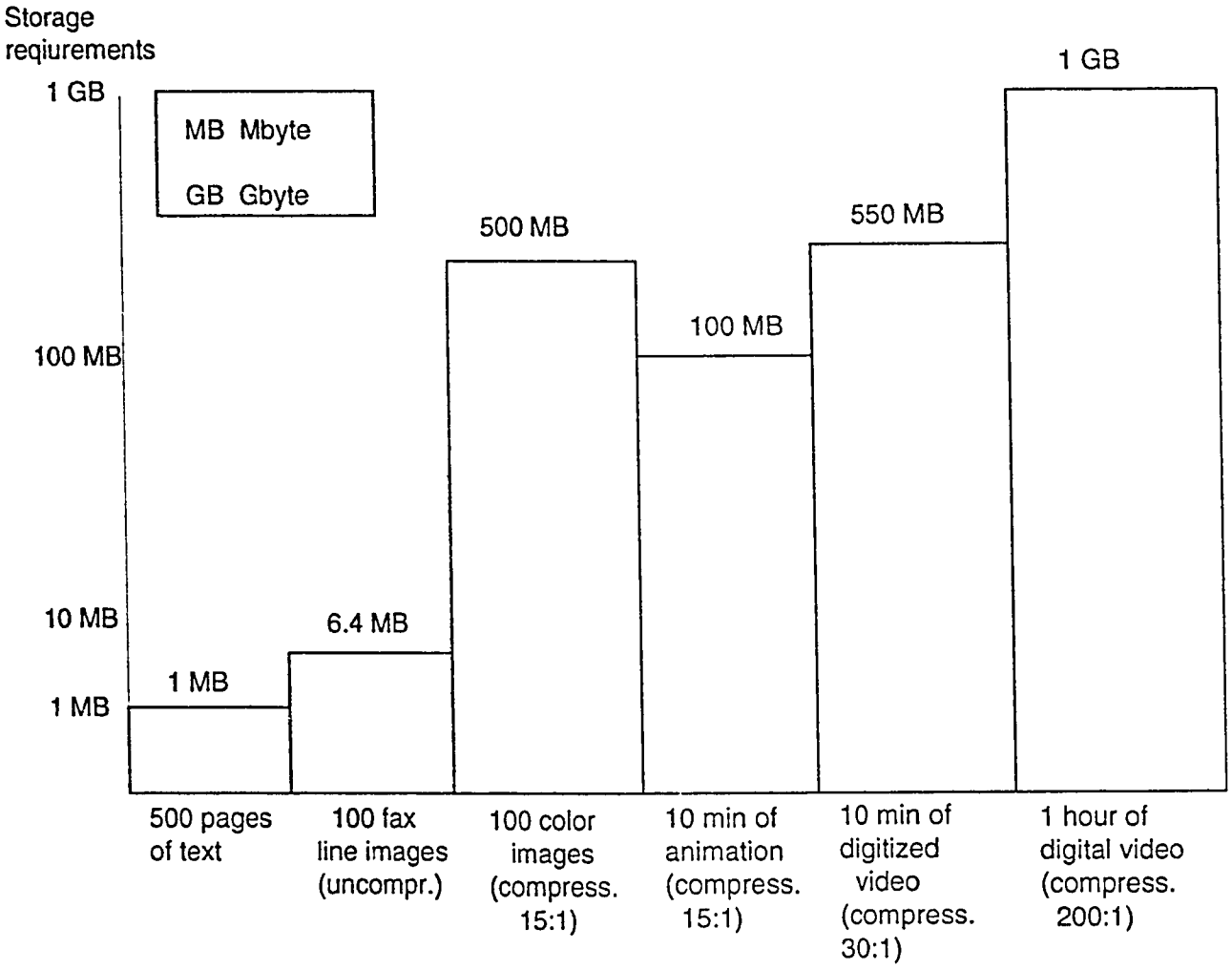


Figure 1.6 : Compressed storage requirements for images and video.

of error is an acceptable trade-off for increased compression [1].

The lossy techniques are prediction-based (Adaptive Differential Pulse Code Modulation-ADPCM), or frequency-based (Discrete Cosine transform-DCT) or importance-based (uses special features of images as basis for compression,i.e., Digital Video Interactive, DVI). Hybrid compression techniques combine several approaches such as DCT and Vector quantization (VQ) or differential pulse code modulation (DPCM)[3].

The image compression techniques studied in this thesis are of lossy type, and the approaches used are based on a system theoretical viewpoint, i.e. as a model reduction problem.

1.3 Literature Survey

The problem of 2-D model reduction has been studied by a number of researchers, however most of the available results on 2- D model reductions are developed for separable systems.

2-D rational transfer functions which are separable in the denominator are simply referred to as separable and the corresponding state space representation given below will have A_2 (or equivalently A_3) $= 0$.

$$\begin{bmatrix} X^h(h+1,k) \\ X^v(h,k+1) \end{bmatrix} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \begin{bmatrix} X^h(h,k) \\ X^v(h,k) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} U(h,k) \qquad (1.1)$$

$$Y(h,k) = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} X^h(h,k) \\ X^v(h,k) \end{bmatrix} \qquad (1.2)$$

where $X^h(h,k)$ is the horizontal state,

$X^v(h,k)$ is the vertical state,

$U(h,k)$ is the input value,

$Y(h,k)$ is the output value,

and constant A_i, B_i, C_i are matrices of suitable dimensions.

A 2-D transfer function is given [5] by:

$$G(z_1,z_2) = \frac{n(z_1,z_2)}{d(z_1,z_2)} = \frac{n(z_1,z_2)}{d_1(z_1)d_2(z_2)} \qquad (1.3)$$

where $n(z_1,z_2)$,$d(z_1,z_2)$,$d_1(z_1)$ and $d_2(z_2)$ are all polynomials in their respective terms.

In contrast to the non-separable case, it is always possible to obtain a minimal realization of a given separable transfer function. A separable realization is minimal if and only if it is both separately locally reachable and locally observable [6]. A separable system is stable iff $A_{(1)}$ and $A_{(4)}$ are stable. In addition, testing for the stability of separable systems only requires two stability tests for two 1-D systems. This is is in contrast to the non-separable case, where the stability test is much more complicated [7].

Separable filters have been utilized in the approximation of 2-D quarter-plane filters, asymmetric half plane filters,etc. They also play an important role in feedback factorizing control of multidimensional systems [8].

In [9], Antoniou, Paraskevopoulos & Varoufakis, considered a minimal state-space realization algorithm for the case of product-factorable-in-denominator or numerator transfer functions. It is shown that the state-space description of the system can be obtained by direct inspection of the circuit diagram realization.

Kumar, Fairman, & Sveinsson [10], presented a separately balanced realization and model reduction technique for quarter-plane causal recursive separable denominator(CRSD) 2-D filters. The algorithm derives a controllable canonic form realization from the given Markov parameters[16], and then proceeds to obtain a reduced-order model whose behavior approximates the given input-output behavior.

Lashgari, Silverman, & Abramatic [11], considered the approximation of CRSD filters and model reduction of Roesser-CRSD models. Two algorithms, one for the approximation and the other for model reduction were presented in [11]. The approximation algorithm obtains two 1-D causal recursive filters from two Hankel matrices that are derived from the given impulse response data. These are then aggregated to obtain the required 2-D CRSD filter. The model reduction algorithm assumes a 2-D CRSD balanced initial model, which can always be obtained using an equivalence transformation from any CRSD model with stable diagonal matrices, and derives a reduced order model. Error analysis and sufficient conditions for

minimality are also given to indicate the validity of the operator.

The minimal realization of all-pole and all-zero 2-D transfer functions is considered by Varoufakis, Parskevopoulos & Antoniou [12]. A simple technique for deriving the state space representation of a given transfer function by inspection from the block diagram is also presented.

A model reduction technique for 2-D separable-denominator transfer function was proposed by Zhou, Li, & Lee [13]. The algorithm uses 1-D balanced truncation and optimal Hankel norm approximation to derive a stable 2-D reduced order model of a stable separable denominator system. Error bounds are also given.

Lin, Kawamata, & Higuchi [14], presented a design procedure for 2-D separable denominator digital filters (SDDF) based on a reduced dimensional decomposition outlined in [15] . In this approach, the given 2-D impulse response specifications are decomposed into a pair of 1-D impulse response specifications which are optimal in the Hankel norm sense. Then 1-D filters using balanced approximation are realized separately for these 1-D specifications. Finally the required 2-D SDDF is obtained by aggregating the 2 reduced-model 1-D filters. Again, error analysis is also presented to assess the accuracy of the approximation. Furthermore, issues about the controllability, observability and minimality of the proposed model are discussed by the same authors in a separate paper [16] .

In [7], 2-D approximation schemes were developed using Schwarz canonical form, which is also known as the Mansour's form. The results developed for 1-D systems

using balanced model reduction have also been generalized for the case of 2-D separable systems. Related references are [7, 13, 11, 10], and [17] .

As it can be inferred from this brief yet concise review above, the problem of 2-D model reduction has already been studied by a number of researchers. In this thesis, we apply the H_∞ norm approximation techniques for 1-D systems to image and speech compression.

1.4 Thesis organization

Chapter 2 gives the Mathematical preliminaries, where 2-D state-space models, different representations of 2-D sequences as 1-D sequences, normed spaces are covered together with some details on the balanced model reduction and the Hankel norm approximation schemes and the H_∞ norm system approximation by identification techniques studied in this thesis.

The compression algorithms developed and implemented are described in Chapter 3. These algorithms are based on the mathematical concepts already introduced in chapter 2 and are outlined in a compact but comprehensive form through the use of necessary notes and explanations. The basics of data compression are also adressed.

In Chapter 4, the results of the research carried out using these algorithms are presented and analyzed for both image and speech compression. Two performance

measures for image compression, namely the subjective (qualitative) and objective(quantitative) image fidelity parameters are also discussed. The performance of the speech compression algorithm is studied using the speech signal quality measure. All of the developed algorithms have been evaluated using standard images and some real speech data.

Finally, conclusions and suggestions for further work are given in Chapter 5.

Chapter 2

Mathematical Preliminaries

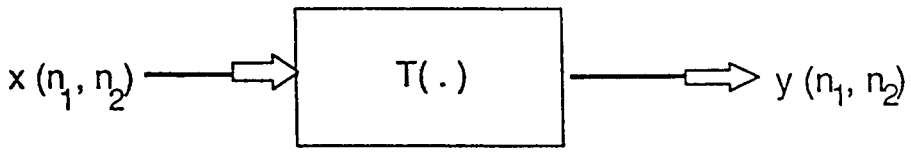
2.1 Introduction

Two-dimensional systems are systems with two independent variables as shown in Figure 2.1. In recent years a number of state-space models [18, 19, 20] for discrete two-dimensional systems have been developed.

State-space realizations of linear systems that are described by transfer functions, difference equations or convolution summations are easily formulated. The model can then be systematically analyzed using many known techniques [18].

Although discrete time state-space models of 2-D systems make both the formulation and analysis of 2-D linear systems manageable, the results so far obtained do not yet constitute a unified theory.

Reduced models of 2-D signal and image systems will make the computational



$$y(n_1, n_2) = T[x(n_1, n_2)]$$

Figure 2.1 : A discrete-time two-dimensional system.

burden lighter while providing reasonably accurate results. Approximating the model of a linear dynamical system by a lower-order one can considerably simplify various analysis and synthesis procedures.

However, most of the work that has been done on 2-D model reduction focusses on separable systems since they are easier to deal with. Approximation schemes for non-separable systems are yet to be developed.

The representation of 2-D sequences by 1-D ones has a number of possible uses; since many well developed and understood 1-D design algorithms and analytical techniques can be used for both the design and analysis of 2-D systems.

This chapter is organized as follows: Different realizations of 2-D systems are given in section 2 while the representation of 2-D sequences as 1-D sequences is presented in section 3. Normed spaces are then reviewed in sections 4 to 7. Balanced model reduction, the Hankel approximation scheme and the H_∞ norm system approximation using identification techniques are covered in sections 8,9, and 10, respectively.

2.2 2-D State-Space Models

The three main two-dimensional discrete state-space models commonly used are:

1. **Roesser's Model** [18, 21]: It is mathematically described by the following equations:

$$\begin{bmatrix} X^h(h+1,k) \\ X^v(h,k+1) \end{bmatrix} = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix} \begin{bmatrix} X^h(h,k) \\ X^v(h,k) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} U(h,k) \qquad (2.1)$$

$$Y(h,k) = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} X^h(h,k) \\ X^v(h,k) \end{bmatrix} \qquad (2.2)$$

where $X^h(h,k)$ is the horizontal state,

$X^v(h,k)$ is the vertical state,

$U(h,k)$ is the input value,

$Y(h,k)$ is the output value,

and constant A_i, B_i, C_i are matrices of suitable dimensions.

Each point (h,k) is tied to a local state vector that is the sum of the horizontal state $X^h(h,k)$ and the vertical one $X^v(h,k)$. The horizontal state is assumed to depend only on the values of X^v and X^h at (h-1,k) while the vertical state on the values of X^v and X^h at (h,k-1) [22, 21].

2. **Atassi's Model**[21]: The local state at (h+1,k+1) is assumed to depend on the state values at (h,k+1), (h+1,k) and (h,k) and on the input value at (h,k) [21].

It is expressed as follows:

$$X(h+1,k+1) = A_1X(h+1,k)+A_2X(h,k+1)-A_1A_2X(h,k)+BU(h,k) \quad (2.3)$$

where the matrices $A_i (i = 1, 2)$ are all constants.

$$Y(h,k) = CX(h,k) \tag{2.4}$$

with $A_1 A_2 = A_2 A_1$.

3. **Fornasini-Marchesini's Model**[21]:

Here the local state at (h+1,k+1) is assumed to depend on the state and input values at (h+1,k) and (h,k+1), i.e,

$$X(h+1,k+1) = A_1 X(h+1,k) + A_2 X(h,k+1) + B_1 U(h+1,k) + B_2 U(h,k+1) \tag{2.5}$$

$$Y(h,k) = CX(h,k) \tag{2.6}$$

The Atassi model is dedicated to the realization of only separable causal rational transfer functions, that is those whose denominator $q(z_1, z_2)$ can be written as a product of 2 polynomials $q_1(z_1)$ and $q_2(z_2)$. The Roesser and Fornasini-Marchesini models realize the whole class of causal rational functions(i.e proper rational functions) regardless of whether their denominators are separable or not. Thus the realization problem of a 2-D rational transfer function can always be solved by models (1) and (3). The Roesser's model is also a particular case of the Fornasini-

Marchesini's model with the vector[21]:

$$X(h,k) = \begin{bmatrix} X^h(h,k) \\ \\ X^v(h,k) \end{bmatrix} \tag{2.7}$$

Eising [20] showed that all these models are special cases of a new model which is a straightforward generalization of the 1-D case.

2.3 Implementation of 2-D sequences as 1-D sequences

Aside from providing a complete understanding of 2-D systems , there are a number of other practical advantages to the representation of 2-D sequences by 1-D ones. It allows for the design and implementation of 2-D systems as 1-D ones. In this way, many of the difficulties associated with 2-D systems such as the lack of a factorization theorem, can be partially circumvented. In addition, there are certain image processing devices such as television cameras and drum scanners which can be conveniently understood in terms of 1-D mappings[23].

The following representations of 2-D signals as 1-D ones have been considered and implemented for a 2-D array of finite duration by mapping it to a 1-D sequence [4, 23]. All the mappings carried out are on a one-to-one basis and can all be readily seen as being invertible.

1. **Lexicographic column ordering**: This is a column ordered vector and is

defined, for an $M \times N$ array , $g(Nm+n) = f(m,n)$, as follows:

$$g^T = [\underbrace{f(1,1)f(2,1)...f(M,1)}_{1^{st}col.}\underbrace{f(1,2)...f(M,2)}_{2^{nd}col.}...\underbrace{f(1,N)...f(M,N)}_{N^{th}col.}]^T \quad (2.8)$$

The samples of the sequence here are formed by concatenating(stacking) each column to the right of the previous column of the array $f(m,n)$.

2. **Lexicographic row ordering:**$g(m+Mn) = f(m,n)$, similar to the above except that the concatenation(stacking) is done row-wise i.e

$$g^T = [\underbrace{f(1,1)f(1,2)...f(1,N)}_{1^{st}row}\underbrace{f(2,1)...f(2,N)}_{2^{nd}row}...\underbrace{f(M,1)...f(M,N)}_{M^{th}row}]^T \quad (2.9)$$

3. **The Zig-Zag ordering:** This mapping is as shown in Figure 2.2 and is defined as follows:

$$g(p) = \begin{cases} m + \sum_{i=0}^{m+n} i & \text{for } m+n \text{ even} \\ n + \sum_{i=0}^{m+n} i & \text{for } m+n \text{ odd} \end{cases} \quad (2.10)$$

At each point, the output intensities,i.e. $g(p)$, are evaluated in the order g(0,0), g(0,1), g(1,0), g(2,0), g(1,1), g(0,3) ... as shown in Figure 2.2 The zig-zag index sequence creates a 1-D vector where lower frequencies tend to be at lower indices.

4. **Manry and Aggarwal ordering** [24]: This ordering scheme is shown in Figure 2.3 and is based on a first-order function defined as:

$$Q(M,N) = N + \sum_{i=o}^{M+N} i \quad (2.11)$$

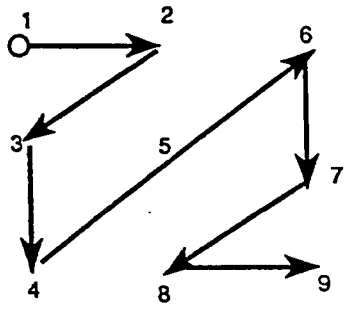


Figure 2.2 : Zig-Zag ordering of 3 X 3 sequence

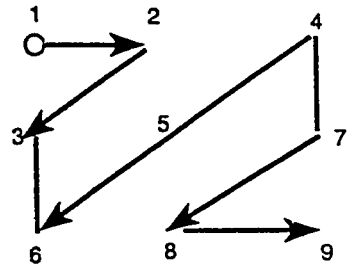


Figure 2.3 : First order function of Manry and Aggarwal for a 3 X 3 sequence

At each point, the output intensities, i.e. $Q(M,N)$ are evaluated in the order g(0,0), g(1,0), g(0,1), g(2,0), g(1,1)... as shown in Figure 2.3.

5. **Lexicographic block-column ordering**: This is similar to 1 above except that here the columns are taken in blocks. The 2-D sequence array is taken in blocks of specified size M_b by N_b where $M_b < M$, $N_b < N$ and concatenated column-wise into one dimensional sequence in a left to right and top to bottom fashion, see Figure 2.4 . The total number of blocks is given by $\left(\frac{M \times N}{N_b \times M_b}\right)$. This is also done for the lexicographic block-row ordering as explained below.

6. **Lexicographic block-row ordering**: This is similar to the above except that here rows are taken in blocks instead of columns.

Note that for each non-negative M and N , $g(p)$ and $Q(M,N)$ in (2.10) and (2.11) respectively equal the number of intensities which are filtered before the intensity f(M,N) is filtered.

Specific representations of 2-D discrete arrays or sequences as 1-D sequences have also been studied by researchers. In [24], Manry and Aggarwal treat the problem of realizing a 2-D recursive filter as a 1-D recursive filter. To this end , the authors introduced 2 integer-valued functions over the set of integers. The purpose of the first function, called the order function , is to describe the sectioning of the 2-D array following a specific scheme, and the purpose of the second function is to describe the scheme of concatenation of these sections to obtain a 1-D array. On the other hand,

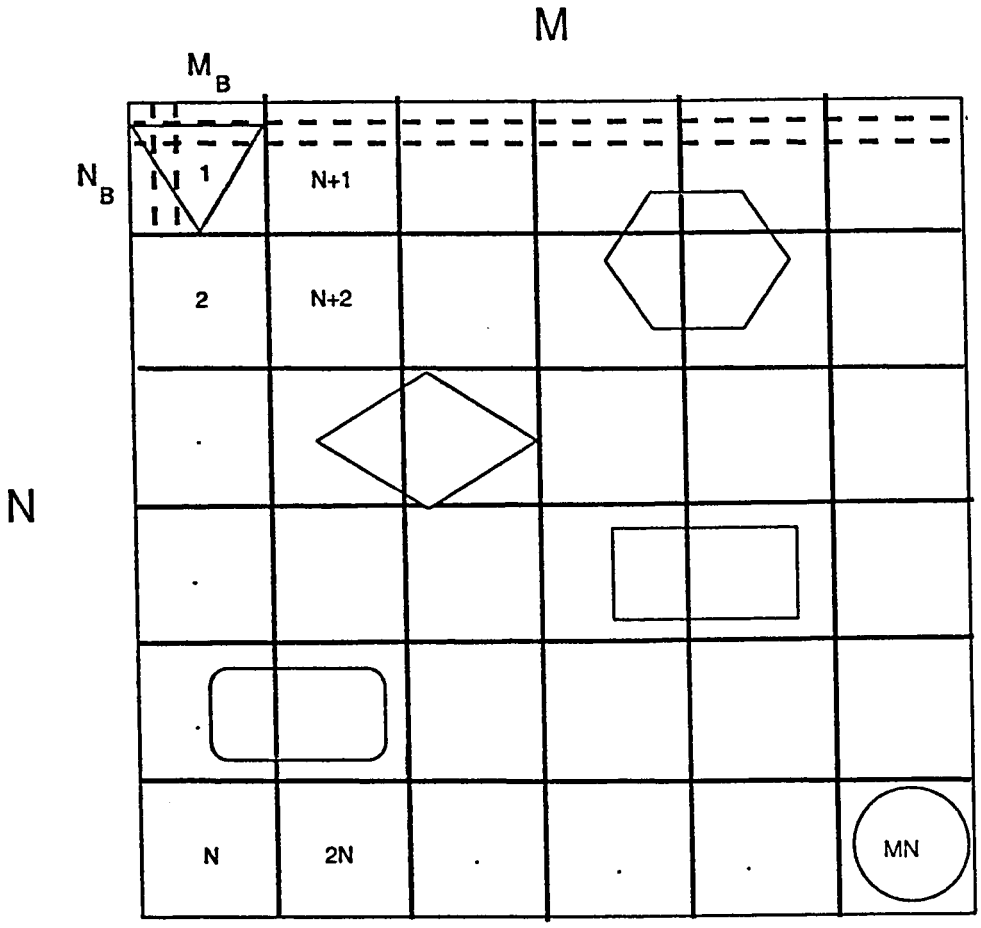


Figure 2.4 : Lexicographic Block-column ordering

Mersereau and Dudgeon [23], limit themselves to the so-called first approximate filter of Manry and Aggarwal and primarily consider the implementation of 2-D non-recursive filters. They develop a 1-D finite-extent MN point sequence called the projection by concatenating the columns of the original 2-D array.

2.4 Norms

2.4.1 Norms for Signals

We consider signals mapping from $(-\infty, \infty)$ to $\Re$. These signals are assumed to be piecewise continuous. Of course, a signal may be zero for $t < 0$ (i.e., it may start at time $t = 0$).

A norm has the following four properties [25]:

1. $\|u\| > 0$

2. $\|u\| = 0 \iff u(t) = 0, \forall t.$

3. $\|au\| = |a|\|u\|, \forall a \in \Re.$

4. $\|u + v\| \leq \|u\| + \|v\|.$

We define the following norms:

1-Norm. The 1-norm of a signal $u(t)$ is the integral of its absolute value [25]:

$$\|u\|_1 := \sum_{-\infty}^{\infty} |u(t)|dt \tag{2.12}$$

2-Norm. The 2-norm of $u(t)$ is [25]:

$$\|u\|_2 := \left(\sum_{-\infty}^{\infty} u(t)^2 dt\right)^{1/2} \tag{2.13}$$

Remark: The *instantenous power* of a signal $u(t)$ is defined to be $u(t)^2$ and its energy is defined as the square of the 2-norm.

∞-Norm. The ∞-norm of a signal $u(t)$, is the least upper bound of its absolute value [25] :

$$\|u\|_\infty := \sup_t |u| \tag{2.14}$$

For example, the ∞-norm of $(1 - e^{-t})1(t)$ equals 1. Here $1(t)$ denotes the step function [25].

2.4.2 Norms for systems

We consider systems that are linear, time-invariant, causal, and (usually) finite dimensional. In the time domain an input-output model for such a system has the form of a convolution equation [25]

$$y = G * u \tag{2.15}$$

that is,

$$y(t) := \sum_{-\infty}^{\infty} G(t - \tau)u(\tau)d\tau \tag{2.16}$$

Let $\hat{G}(s)$ denote the transfer function, the laplace transform of G. We introduce two norms for the transfer function $\hat{G}$.

2-Norm.

$$\|\hat{G}\|_2 := \left(\frac{1}{2\pi}\int_{-\infty}^{\infty}|\hat{G}(jw)|^2 dw\right)^{1/2} \tag{2.17}$$

∞-Norm.

$$\|\hat{G}\|_\infty := \sup_{\omega}|\hat{G}(jw)| \tag{2.18}$$

Remark: The ∞-norm of $\hat{G}$ equals the distance in the complex plane from the origin to the farthest point on the Nyquist plot of $\hat{G}$. It also appears as the peak value on the Bode magnitude plot of $\hat{G}$ [25].

Note:

The Norms given above are for single-input single-output (SISO) cases. However, they are also defined for Mutliple-input Multi-output (MIMO) cases as follows:

2-Norm.

$$\|\hat{G}\|_2 := \left(\frac{1}{2\pi}\int_{-\infty}^{\infty} trace(G(jw)G^*(jw))dw\right)^{1/2} \tag{2.19}$$

∞-Norm.

$$\|\hat{G}\|_\infty := \sup_{\omega}\sigma_{max}|\hat{G}(jw)| \tag{2.20}$$

2.5 Singular Values

The singular values of a matrix $A \in C^{m\times n}$ are given by [26]:

$$\sigma_i = [\lambda_i(A^*A)]^{1/2} \tag{2.21}$$

where λ_i are the eigenvalues of A^*A for $i = 1$ to $\min(m, n)$, and the non-negative square-root is taken. It can be shown that

$$\|A\|_2 = \sigma_{max}(A) \tag{2.22}$$

The spectral radius of A is given by

$$\rho(A) = max|\lambda_i(A)| \tag{2.23}$$

For any $m \times n$ matrix A, there exists a singular value decomposition(SVD) given by

$$A = U\Sigma V^T \tag{2.24}$$

where Σ is an $m \times n$ matrix defined by

$$\Sigma_{ij} = \begin{cases} \sigma_i & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{2.25}$$

where $\sigma_i \geq \sigma_{i+1} \geq \sigma_{i+2} ... \geq 0$ and U and V are unitary matrices of dimension $m \times m$ and $n \times n$ [26], respectively.

2.6 The H_∞ Hardy Space

This consists of all complex-valued functions $F(s)$ of a complex variable s which are analytic and bounded in the open right half-plane, Re $s > 0$; where

"bounded" means that there is a real number b such thatn [25]

$$|F(s)| \leq b, Re\ s > 0 \tag{2.26}$$

The least such bound b is the H_∞-norm of F, denoted by $\|F\|_\infty$.

Equivalently,

$$\|F\|_\infty := \sup_w |F(jw)| : w \in \Re \tag{2.27}$$

The H_∞-norm of a transfer function matrix, is the maximum over all frequencies of its largest singular value.

2.7 Balanced approximation Method

The balanced approximation method of model reduction initiated by Moore [27], proved to be very significant because of its desirable properties, such as good error bounds, computational simplicity, stability and its close connection to robust multivariable control [27]. Extensive studies on balanced representations have been undertaken for the determination of reduced order models which contain only the most controllable and most observable states of the system. The main idea behind balancing is that there exists state-space coordinates where the controllability and observability gramians are equal and diagonal. The diagonal entries of the gramians are called the Hankel singular

values of the system and they provide a measure of *how* controllable and observable a state is. A natural way to achieve model reduction is to keep only the most controllable and most observable states corresponding to the largest Hankel singular values of the system.

2.7.1 Continuous-Time Systems

In this subsection, continuous-time balanced model reduction will be reviewed.

Assume that the system

$$\dot{x} = Ax(t) + Bu(t) \tag{2.28}$$

$$y(t) = Cx(t) \tag{2.29}$$

where $A \in \Re^{n\times n}$, $P \in \Re^{n\times m}$,and $C \in \Re^{p\times m}$ is asymptotically stable, controllable and observable. The controllability gramian P and observability gramian Q are defined as

$$P = \int_0^\infty e^{At} BB' e^{A't} dt \tag{2.30}$$

$$Q = \int_0^\infty e^{A't} C'C e^{At} dt \tag{2.31}$$

and can be found as the unique positive definite solution of the following Lyapunov equations:

$$AP + PA' + BB' = 0 \tag{2.32}$$

$$A'Q + QA + C'C = 0 \tag{2.33}$$

where ($'$) denotes the transposition operation. If the state-space representation of the system (A, B, C) is transformed to another representation using a non-singular transformation T, then, a new state-space representation of the system $(\bar{A}, \bar{B}, \bar{C})$ is obtained where

$$\bar{A} = T^{-1}AT, \bar{B} = T^{-1}B, \bar{C} = CT \tag{2.34}$$

and the gramians will be transformed to

$$\bar{P} = T^{-1}PT'^{-1}, \bar{Q} = T'QT, \overline{PQ} = T^{-1}PQT \tag{2.35}$$

Thus, the gramians P and Q depend on the state-space co-ordinates. However, the eigenvalues of their products PQ are invariant under state-space transformations and are input/output invariant. The square root of the eigenvalues of PQ are called the Hankel singular values of the system $G(s) = C(sI - A)^{-1}B$ and they are denoted by σ_i where $\sigma_i = [\lambda_i(PQ)]^{1/2}$ and $\lambda_i(PQ)$ is the i'th eigenvalue of PQ.

A realization $(\bar{A}, \bar{B}, \bar{C})$ of $G(s)$ is said to be balanced, if $\bar{P} = \bar{Q} = \sum$ where $\Sigma = diag(\sigma_1 \geq \sigma_2 \geq ..... \geq \sigma_n) \geq 0$. The states of such a representation are then balanced between controllability and observability. Thus, they represent a convenient structure for model reduction since any states having weak controllability and weak observability can be neglected without causing any imbalance in controllability or in observability properties of the remaining states [27].

Let the balanced system be partitioned as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u \tag{2.36}$$

$$y = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{2.37}$$

where $x_1 \in \Re^r$ contains the most controllable and most observable states and $x_2 \in \Re^{n-r}$ contains the least controllable and least observable states. Also, let Σ be partitioned compatibly as

$$\Sigma = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \tag{2.38}$$

where $\Sigma_1 = diag(\sigma_1 \geq \sigma_2 \geq ..... \geq \sigma_r)$ and $\Sigma_2 = diag(\sigma_{r+1} \geq \sigma_{r+2} \geq ..... \geq \sigma_n)$. If $\frac{\sigma_r}{\sigma_{r+1}} \gg 1$, then the subsystem

$$\dot{\tilde{x}}_1 = A_{11}\tilde{x}_1 + B_1 u \tag{2.39}$$

$$y_r = C_1\tilde{x}_1 \tag{2.40}$$

is taken as the reduced order model of the full-order balanced system and will contain the most controllable and most observable parts of the system [27].

2.7.2 Discrete-Time Systems

A review of the discrete-time counterparts of the results discussed in the previous subsection will be given here. Although the analysis of discrete-time balanced systems is similar to that of their continuous-time counterparts, there are however some major differences between the two.

Assume that the following discrete-time system is asymptotically stable, reachable and observable

$$x(k+1) = Ax(k) + Bu(k) \tag{2.41}$$

$$y(k) = Cx(k) \tag{2.42}$$

The reachability gramian P and observability gramian Q are defined as

$$P = \sum_{0}^{\infty} (A)^k BB' (A')^k \tag{2.43}$$

$$Q = \sum_{0}^{\infty} (A')^k C'CA^k \tag{2.44}$$

and can be found as the unique positive-definite solution of the following Lyapunov equations:

$$APA' - P + BB' = 0 \tag{2.45}$$

$$A'QA - Q + C'C = 0 \tag{2.46}$$

As in the continuous-time case, it is possible to find a transformation matrix which makes the two gramians both diagonal and equal (say to Σ). The

rest of the analysis required to obtain the reduced order model is completely analogous to that of the continuous-time case; and it will therefore not be repeated here.

2.8 Hankel Approximation Scheme

Hankel Approximation is developed in [28, 29]. An optimal Hankel-norm reduction algorithm is given by Glover in [29] for continuous-time systems and is reviewed below. This algorithm can also be used for discrete-time systems through a bilinear transformation.

There is an rth order stable system

$$\hat{G}_r(z) = \hat{C}(zI - \hat{A})^{-1}\hat{B} + \hat{D} \tag{2.47}$$

such that

$$\|G(z) - \hat{G}_r(z)\|_\infty < \sum_{i=r+1}^{n} \sigma_i \tag{2.48}$$

where $G(z)$ is the transfer matrix, $G_r(z)$ = reduced order transfer matrix and σ_i = ith Hankel singular value. The procedure is outlined below:

1. Obtain a balanced realization of $G : (A, B, C, D)$ namely one such that the Lyapunov equations:

$$A\Sigma + \Sigma A^T + BB^T = 0 \tag{2.49}$$

and

$$A^T\Sigma + \Sigma A + CC^T = 0 \tag{2.50}$$

have a common and diagonal solution $\Sigma = \begin{bmatrix} \sigma_1 I & 0 \\ 0 & \Sigma_1 \end{bmatrix}$ where $\Sigma_1 = diag(\sigma_2, ..., \sigma_n)$, and $\sigma_1 > \sigma_2 \geq ... \geq \sigma_n > 0$. Note that $|G|_H = \sigma_1$.

2. Partition A, B, and C conformally with Σ thus:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, B_1 = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, C = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \tag{2.51}$$

Let the Hankel operator, $\Gamma = \Sigma_1^2 - \sigma_1^2 I$, where the unit matrix has the same dimension as Σ_1 - not usually the same as the unit matrix appearing in Σ, and choose a U such that $UU^T = I$ and $B_1 = -C_1^T U$; such a U always exist, since $B_1 B_1^T = C_1^T C_1$.

3. Define the following matrices :

$$\hat{A} = \Gamma^{-1}(\sigma_1^2 A_{22}^T + \Sigma_1 A_{22} \Sigma_1 - \sigma_1 C_2^T U B_2^T) \tag{2.52}$$

$$\hat{B} = \Gamma^{-1}(\Sigma_1 B_2 + \sigma_1 C_2^T U) \tag{2.53}$$

$$\hat{C} = -C_2 \Sigma_1 - \sigma_1 U B_2^T \tag{2.54}$$

$$\hat{D} = -D + \sigma_1 U \tag{2.55}$$

4. Use bilinear transformation to get $\hat{G}(z)$

for the approximated system.

Note that this $\hat{D}$ will give $\|G(z) - \hat{G}(z)\|_\infty < 2(\sum_{i=r+1}^{n} \sigma_i)$. A constant matrix D_r is computed in [29], which decreases the upperbound above to half the sum of the tails.

2.9 Least-Squares(LS) Identification

The Least-squares(LS) technique attempts to fit a given data to a given model by minimizing the sum of the squared errors between the model output and the observations (i.e. data), assuming an Auto-Regressive (ARX) model with exogenous input given by

$$A(q^{-1})y(t) = B(q^{-1})u(t) + e(t) \tag{2.56}$$

which represents the difference equation of the input/output model with

$$A(q^{-1}) := 1 + a_1 q^{-1} + ... + a_r q^{-r} \tag{2.57}$$

and

$$B(q^{-1}) := b_1 q^{-1} + ... + b_r q^{-r} \tag{2.58}$$

where $u(t)$ and $y(t)$ respectively denote the system input and output and r denotes the model order. Further $e(t)$ is assumed to be a sequence of independent and identically distributed (i.i.d.) random variables (white noise) and q^{-1} is the backward shift operator. Defining the following parameter and data vectors θ and $\phi(t)$ respectively as

$$\theta := (a_1, ..., a_r, b_1, ..., b_r)^T \tag{2.59}$$

$$\phi(t) := (-y(t-1), ..., -y(t-r), u(t-1), ..., u(t-r))^T \tag{2.60}$$

(2.57) can be written as

$$y(t) = \phi^T(t)\theta + e(t) \tag{2.61}$$

The LS estimate of θ is defined as the vector $\hat{\theta}_{LS}$ that minimizes the loss function

$$V_{LS}(\theta) = \sum_{t=1}^{N} \epsilon_{LS}^2(t, \theta) \tag{2.62}$$

where N is the number of data points and $\epsilon(t, \theta)$ is called the residual which is defined thus

$$\epsilon_{LS}(t, \theta) := y(t) - \phi^T(t)\theta \tag{2.63}$$

Unlike other identification methods(i.e., output-error), $\hat{\theta}_{LS}$ can be obtained by a non-iterative closed-form equation as

$$\hat{\theta}_{LS} = \left[\sum_{t=1}^{N} \phi(t)\phi^T(t)\right]^{-1} \left[\sum_{t=1}^{N} \phi(t)y(t)\right] \tag{2.64}$$

From the identification literature, it is established that the LS method produces biased and inconsistent estimates in the presence of measurement noise. Inspite of this, the LS method has two very attractive properties which has resulted in its usefulness in signal processing applications. These are: (1) LS identification is non-iterative and hence is computationally extremely fast, and (2) it always converges to a global minimum since it is a linear estimator with a quadratic convergence criterion. Another interesting property of the LS estimator is that the stability of the estimated model is ensured when the input data is white. The last property makes the LS estimation very popular in spectrum estimation [30].

2.10 H_∞ Norm System Approximation using Identification Techniques

2.10.1 The H_∞ Norm Model Reduction Problem

In general, model reduction is the approximation, with respect to a given criterion, of a given transfer function with n modes by one with $r \ll n$ modes. In other words, a higher order system is approximated by a lower order one. The optimal H_∞ norm model reduction problem which is also known in mathematics literature as the complex rational Chebychev approximation is defined as

follows:

Given a transfer function $g_n(z) \in RH_\infty$ with degree n, find a transfer function $g_r(z)$ with degree r such that $\|g_n - g_r\|_\infty$ is minimized.

It has been shown that a not necessarily unique solution exists for the optimal H_∞ norm approximation problem. In [31], D. Kavranoglu and M. Bettayeb gave a state-space characterization of the solution to this problem.

2.10.2 Application of Least-Squares Method

An inherent drawback of state-space based methods is that when one obtains a balanced realization of an n'th order system, one needs $n^2+2n+1 = (n+1)^2$ coefficients, while in the transfer function form only $(2n+1)$ coefficients are needed. This (i.e. the state-space approach) is thus prohibitory for very high order systems. Therefore, the need for developing direct frequency domain approximation schemes that do not depend on state-space tools without compromising on the H_∞ norm of the approximation error exists.

Given an nth order stable system transfer function with a reduced rth order model respectively as

$$G_n(z) = \frac{\beta_0 + \beta_1 z^{-1} + ... + \beta_n z^{-n}}{1 + \alpha_1 z^{-1} + ... + \alpha_n z^{-n}} \tag{2.65}$$

and

$$G_r(z,\theta) = \frac{b_0 + b_1 z^{-1} + ... + b_r z^{-r}}{1 + a_1 z^{-1} + ... + a_r z^{-r}} \tag{2.66}$$

where $r < n$ and $\theta := (a_1, ..., a_r, b_1, ..., b_r)^T$. The vector parameter θ of the optimal reduced-order model for a specified r is given as

$$\theta_{opt} = arg \min_{\theta} \|G_n(.) - G_r(., \theta)\|_\infty \tag{2.67}$$

A solution to the above H_∞ model reduction problem is given in [32] by D. Kavranoglu and M.S. Ahmed in which they derived the solution of the problem of H_∞ approximation by identification, based on weighted output error which was further modified to use LS. The intituive appeal of their solution is still retained with this modification , since in each iteration the new input to the system and model would be obtained such that the input spectrum is similar to that of the output error between the system and model during the preceding iteration. Hence a large output error at a particular frequency ω_0 in one iteration would increase the energy of the input at the same frequency, ω_0. The LS identification method attempts to minimize the sum of the squared residuals and in so doing also reduces the output error at ω_0 due to the concentration of input energy at this frequency. The authors solution in [32] provides an output error weighting of the least squares criterion function. They also found out that the resulting reduced-error model based on LS is better than the one based on output error(OE). This was due to the guaranteed convergence to

the global minimum of their modified LS criterion function.

2.11 Nehari's Theorem

Lemma 1 [33]: Given an $m \times m$, n'th order transfer function $G_n(z)$, let $G_r(z)$ be an r'th order transfer function. We then have :

$$\sigma_{r+1} \leq \|G_n - G_r\|_\infty \tag{2.68}$$

where σ_{r+1} is the $r+1$'st Hankel singular value of $G_n(z)$. This lemma provides a very valuable lower bound for H_∞ approximation. It is also well known that the H_∞ norm of the approximation error for balanced model reduction is bounded by $2(\sigma_{r+1} + \sigma_{r+2} + \sigma_{r+3} + ... + \sigma_n)$ while for the Hankel norm approximation this error is bounded by $(\sigma_{r+1} + \sigma_{r+2} + \sigma_{r+3} + ... + \sigma_n)$ [29].

2.12 Discrete Cosine Transform

The 1-D DCT pair(i.e. analysis and synthesis) of an N-point sequence $x(n)$, that has values zero outside $0 \leq n \leq N-1$ is defined as follows

$$C_x(k) = \begin{cases} \sum_{n=0}^{N-1} 2x(n) Cos\frac{\pi}{2N} k(2n+1) & 0 \leq k \leq N-1 \\ 0 & \text{otherwise} \end{cases} \tag{2.69}$$

$$x_n = \begin{cases} \frac{1}{N}\sum_{K=0}^{N-1} w(K) Cos \frac{\pi}{2N} k(2n+1) & 0 \leq n \leq N-1 \\ 0 & \text{otherwise} \end{cases} \tag{2.70}$$

In like manner, the 2-D DCT pair for an $N_1 \times N_2$ point sequence $x(n_1, n_2)$ that has values zeros outside $0 \leq n_1 \leq N_1 - 1$ and $0 \leq n_2 \leq N_2 - 1$ is given as follows

$$\begin{aligned} C_x(k_1, k_2) &= \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} \cos \frac{\pi}{2N_1} k_1(2n_1+1) \cos \frac{\pi}{2N_2} k_2(2n_2+1) \\ &\quad for\ 0 \leq k_1 \leq N_1 - 1, 0 \leq k_2 \leq N_2 - 1 \\ &= 0,\ otherwise \end{aligned} \tag{2.71}$$

$$\begin{aligned} x(n_1, n_2) &= \frac{1}{N_1 N_2} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} w_1(k_1) w_2(k_2) C_x(k_1, k_2) \cos \frac{\pi}{2N_1} k_1(2n_1+1) \times \\ &\quad cos \frac{\pi}{2N_2} k_2(2n_2+1) \\ &\quad for\ 0 \leq n_1 \leq N_1 - 1, 0 \leq n_2 \leq N_2 - 1 \\ &= 0,\ otherwise \end{aligned} \tag{2.72}$$

A real $N_1 \times N_2$ point sequence is represented by $N_1 N_2$ real DCT coefficients while a complex $N_1 \times N_2$ point sequence will be by $N_1 N_2$ complex DCT coefficients. The DCT is used primarily in image and signal coding applications for reasons discussed in Chapter 3.

Chapter 3

H_∞-Based Data Compression of Image and Speech Signals

3.1 Introduction

The discrete cosine transform (DCT) is a popular image data compression technique. It has also evolved as the international standard (JPEG-Joint Photographic Expert Group) for still-image data compression. Table 3.1 shows the drawbacks of some compression methods. This choice of the DCT is attributed to the following reasons [1, 4, 5, 34]:

1. Excellent energy compaction property for highly correlated data: compaction of most energy into a fraction of the transform coefficients allows

Table 3.1: Drawbacks of some compression methods.

Methods	Comments
Bit Plane Coding (BPC)	☐ Very sensitive to channel errors. ☐ Requires significant bit-planes to be protected. ☐ Low compression ratio (1-3).
Delta Modulation (DM)	☐ Slope overload leads to blurred object edges. ☐ Grainy or granular noise occurs. ☐ Instability to channel errors. ☐ Low compression ratio (1-3).
Differential Pulse Code Modulation (DPCM),	☐ Auto-correlation computation is difficult. ☐ Sensitive to image data statistics. ☐ Channel errors are cumulative and degrade image quality. ☐ Low compression ratio (1-2). ☐ No closed or explicit form solution for the reconstruction level. ☐ Increased quantizer complexity .
Vector Quantization (VQ)	☐ Possible image patterns can be prohibitvely large. e.g. for an image of size 128 X128 , with 8 bits/pixel , the number of possible patterns = $2.^{2048}$
Discrete Cosine Transform (DCT)	☐ Hardware complexity is high. ☐ Blocking artifact results at times.

us to discard many coefficients without seriously affecting the image.

2. Superior information packing ability: The DCT is a transformation that redistributes or packs most of the information into as close to the fewest possible coefficients thus providing the best sub-image approximations and consequently the smallest reconstruction errors.

3. Good compromise between information packing ability and computational complexity.

4. Fast implementation via Fast-Fourier Transform (FFT).

Digital data compression relies on various computational algorithms implemented either in hardware or software. Compression techniques fall into two broad categories: loss-less, i.e., information-preserving and lossy. In the former catergory, an image can be compressed and decompressed without losing information. In the latter category higher levels of data reduction are provided but result in a less-than-perfect reproduction of the original image. Lossy image compression is useful in applications such as broadcast tv, video conferencing, and facsimile transmission, in which a certain amount of error is an acceptable trade-off for increased compression [1].

The algorithms developed and implemented in this study are all of the lossy category. In this chapter, section 2 outlines the basics of data compression and section 3 covers the DCT-based compression algorithm implemented. The Bal-

anced model reduction based algorithm for the sorted and non-sorted case are presented in subsections 4.1 and 4.2, Hankel norm approximation are presented in subsections 4.3 and 4.4. In subsection 4.5, the H_∞ norm image approximation and in subsection 4.6 the H_∞ norm speech compression algorithms are presented.

3.2 Data Compression Fundamentals

Image data compression addresses the problem of reducing the amount of data required to represent a digital image with a view to reducing its storage requirement and transmission bit rate etc. The underlying basis of the reduction process is the removal of redundant data present in the data. From a mathematical viewpoint, this amounts to conerting a two-dimensional pixel array into a statistically uncorrelated data set. The transformation is applied prior to storage or transmission of the image. At some later time, the compressed image is decompressed to reconstruct the original image or an approximation to it [1].

Image data compression is achieved when redundancies in data are eliminated or reduced. Data redundancy is defined as thus : if n_1 & n_2 denote the number of information-carrying units in two data sets that represent the same information, the relative data redundancy R_D of the first data set (the one

characterized by n_1) is given as :

$$R_D = 1 - \frac{1}{C_R} \tag{3.1}$$

where C_R is the compression ratio, given by $C_R = \frac{n_1}{n_2}$.

For the case $n_2 = n_1$, $C_R = 1$ & $R_D = 0$ indicating (that relative to the the second data set) the first representation of the information contains no redundant data. While $n_2 \ll n_1$, $C_R \to \infty$ & $R_D \to 1$ implying significant and highly redundant data and therefore the possibility of using a high compression ratio. In the final case, $n_2 \gg n_1$, $C_R \to 0$ and $R_D \to -\infty$ indicating that the second data set contains much more data than the original representation. This, of course represents the undesirable case of data expansion. In general, C_R & R_D lie in the open intervals $(0, \infty)$ and $(-\infty, 1)$ respectively [1].

A practical compression ratio, such as 10 or (10:1) means that the first data set has 10 information-carrying units (say bits) for every 1 unit in the second or compressed data set. The corresponding redundancy of 0.9 implies that 90 % of the data in the first data set is redundant [1].

3.3 Algorithm 1: Discrete Cosine Transform-Based Image Compression

1. Sub-divide the given $N \times M$ input image into sub-images of sizes $p \times q$,

hence a total of $\frac{M \times N}{p \times q}$ sub-images.

2. Transform the sub-images to generate $\frac{M \times N}{p \times q}$ transform arrays i.e., compute the two-dimensional (2-D) DCT of each $p \times q$ sub-image block.

3. Map the $\frac{M \times N}{p \times q}$ transform arrays from the 2-D space to 1-D space by concatenating them according to some desired manner (column-wise,row-wise, etc). Note that this mapping is invertible.

4. Compress the DCT coefficients obtained as follows:

   (a) row concatenate the DCT coefficients of each sub-image block from left to right and top to bottom.

   (b) sort the DCT coefficients in an ascending order .

   (c) discard the DCT coefficients obtained in (b) of each $p \times q$ sub-image block according to the specified compression ratio factor, R .

   (d) Restore the DCT coefficients obtained in (c) to their previous order as in (a). This is the reverse of (b).

   (e) inverse row concatenate the elements to reconstruct a reduced 2-D sequence of coefficients from the 1-D sequence obtained in (d). This is also the reverse of (a).

5. Inverse transform the transform-domain blocks i.e compute the inverse 2-D DCT of each $p \times q$ transform-domain block obtained in 4(e).

6. Re-map the $(\frac{M \times N}{p \times q})$ from the 1-D space to the 2-D space. This is the

inverse of (3).

7. Display the compressed image alongside the original image for comparision purposes.

Note:

The goal in step (2) above is to de-correlate the pixels of each sub-image , or to pack as much information as possible into the smallest number of transform coefficients.

In step 4(b), the threshold is inherently adaptive in nature in the sense that the range of transform coefficients for each sub-image vary from one sub-image to another. The underlying concept is that for any sub-image the transform coefficients of largest amplitudes make the most significant contribution to the reconstructed image quality. Because the values of the maximum coefficients vary from one sub-image to another due to the fact that the image contents change from block to block, using a fixed threshold would thus be inefficient.

For step 4(c), the truncated coefficients are obtained by selectively eliminating the DCT coefficients that carry the least information. These coefficients have the smallest impact on reconstructed image quality, for example if $R = 2$ then half of the coefficients would have to be discarded.

3.4 Development of Several H_∞-Based Compression Algorithms

3.4.1 Algorithm 2: Balanced Image Model Reduction

1. Divide the given $N \times M$ image with impulse response matrix G into smaller rectangular blocks each of size $p \times q$. Thus the image is divided into $\frac{NM}{pq}$ blocks. (See Figure 3.1).

2. Map the blocks to a one-dimensional array with impulse response matrix $\tilde{G}$ by concatenating the blocks according to some desired manner (column-wise, row-wise, etc) implying transforming the $N \times M$ image $\rightarrow \frac{NM}{pq}$. Note that this mapping from the 2-D space to 1-D space is invertible. (See Figure 3.2).

3. Calculate the spectral norm of each block. i.e $\|G_i\| \rightarrow (\alpha_1, \alpha_2, ...\alpha_n)$.

4. Obtain a state-space realization of $\tilde{G}$.

5. Find a minimal realization in (4) if it is not already minimal.

6. Approximate this minimal realization by the balanced model reduction method to obtain an approximate $\hat{G}_r$.

7. Synthesize (decompress) $\hat{G}_r$ i.e. find the response of the system using $Y_k = CA^{k-1}B$ where the parameters (A, B, C) are for the approximated

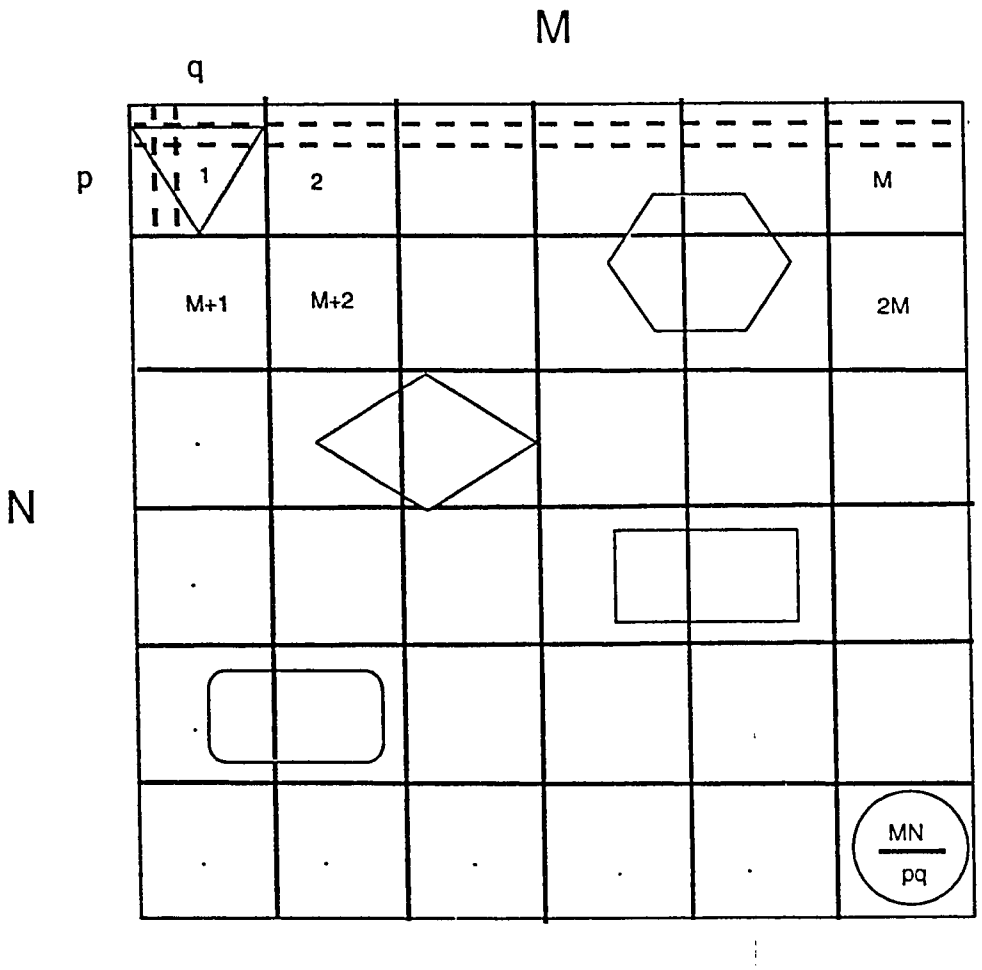


Figure 3.1 : Sub-dividing the original image

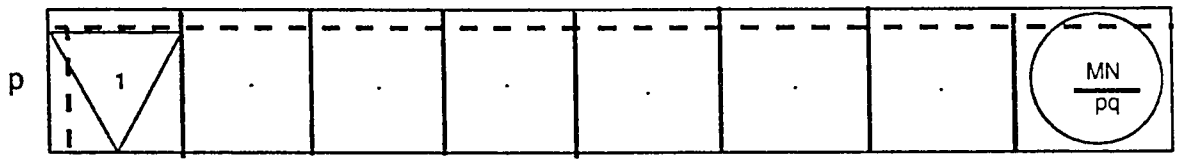


Figure 3. 2: Mapping the sub-divided image blocks

values, i.e.,(A_r, B_r, C_r). Note that this is akin to step 9 of Algrotihm 1.

8. Display the original image G alongside the synthesized approximated image $\hat{G}_r$ for comparison purposes.

3.4.2 Algorithm 3: Sorted-Balanced Image Model Reduction

1. Divide the given $M \times N$ image with impulse response matrix G into smaller rectangular blocks each of size $p \times q$. Thus the image is divided into $\frac{NM}{pq}$ blocks. (See Figure 3.1.)

2. Map the blocks to a one-dimensional array with impulse response matrix $\tilde{G}$ by concatenating the blocks according to some desired manner (column-wise, row-wise, etc) implying transforming the $N \times M$ image $\rightarrow \frac{NM}{pq}$. Note that this mapping from the 2-D space to 1-D space is invertible. (See Figure 3.2).

3. Calculate the norm of each block. i.e., spectral $\|G_i\|_s \rightarrow (a_1, a_2, ...a_n)$.

4. Find the ordering from largest to smallest norms of the $\frac{NM}{pq}$ blocks. i.e $J = (J_1, J_2, ...J_n)$ where J_i are the block indices.

5. Shuffle the 1-D sequence $\tilde{G}$ according to the sorted indices in (4) to obtain $\tilde{G}_s$.

6. Obtain a state - space realization of $\tilde{G}_s$ as shown below:

   For a given $G(z)$[35]:

$$G(z) = F_0 + F_1 z^{-1} + F_2 z^{-2} + ... + F_N z^{-N}$$

   one possible representation of the state-space model parameters are

$$A = \begin{bmatrix} 0_{m\times m} & \cdots & \cdots & 0_{m\times m} \\ I_{m\times m} & 0_{m\times m} & \cdots & . \\ 0_{m\times m} & I_m & \cdots & . \\ 0_{m\times m} & & I_{m\times m} & 0_{m\times m} \end{bmatrix}, B = \begin{bmatrix} I_m \\ 0_{m\times m} \\ . \\ . \\ 0_{m\times m} \end{bmatrix},$$

$$C = \begin{bmatrix} F_1 & F_2 & . & F_N \end{bmatrix}, D = \begin{bmatrix} F_0 \end{bmatrix}$$

   where the impulse response is given by $Y_k = CA^{k-1}B$.

   Note that for every realizable transfer function matrix, the state-space realization is non-unique. A major problem is to find a "good" realization. It is clear that a dynamical-equation realization with the smallest possible dimension is a good realization . Such a realization is called a minimal realization and it is both controllable and observable [35].

7. Find a minimal realization in (6) if it is not already minimal.

8. Approximate this minimal realization by the balanced model reduction method to obtain an approximate $\hat{G}_r$.

9. Synthesize (decompress) $\hat{G}_r$ i.e. find the response of the system using $Y_k = CA^{k-1}B$ where the parameters (A, B, C) are for the approximated values, i.e.,(A_r, B_r, C_r). Note that the approximate system will be typically an infinite impulse response (IIR). Truncate the response by any desired window of interest.

10. Reorder the synthesized image impulse response matrix in (9) according to the sorting indices in (4) to obtain the approximate image $\hat{G}_r$. This is actually the reverse of (4).

11. Display the original image G alongside the synthesized approximated image $\hat{G}_r$ for comparison purposes.

3.4.3 Algorithm 4: Hankel Image Model Reduction

1. Divide the given $N \times M$ image with impulse response matrix G into smaller rectangular blocks each of size $p \times q$. Thus the image is divided into $\frac{NM}{pq}$ blocks.

2. Map the blocks to a one-dimensional space with impulse response matrix $\tilde{G}$ by concatenating the blocks according to some desired manner (column-wise, row-wise, etc) implying transforming the $N \times M \rightarrow p\frac{NM}{pq}$.

3. Calculate the spectral norm of each block. i.e $\|G_i\| \rightarrow (\alpha_1, \alpha_2, ...\alpha_n)$.

4. Obtain a state-space realization of $\tilde{G}$.

5. Find a minimal realization in (4) if it is not already minimal.

6. Approximate this minimal realization by the Hankel model reduction method to obtain an approximate $\hat{G}_r$.

7. Synthesize (decompress) $\hat{G}_r$ using $Y_k = CA^{k-1}B$ where the parameters (A, B, C) are for the approximated values, i.e.,(A_r, B_r, C_r) . Note that the approximate system will be typically an infinite impulse response (IIR). Truncate the response by any desired window of interest.

8. Display the original image G alongside the synthesized approximated image $\hat{G}_r$ for comparsion purposes.

3.4.4 Algorithm 5: Sorted-Hankel Image Model Reduction

1. Divide the given $N \times M$ image with impulse response matrix G into smaller rectangular blocks each of size $p \times q$. Thus the image is divided into $\frac{NM}{pq}$ blocks.

2. Map the blocks to a one-dimensional space with impulse response matrix $\tilde{G}$ by concatenating the blocks according to some desired manner (column-wise, row-wise, etc) implying transforming the $N \times M$ image $\rightarrow \frac{NM}{pq}$.

3. Calculate the spectral norm of each block. i.e $\|G_i\| \rightarrow (\alpha_1, \alpha_2, ...\alpha_n)$.

4. Find the ordering from largest to smallest norms of the $\frac{NM}{pq}$ blocks. i.e $J = (J_1, J_2, ...J_n)$ where J_i are the block indices.

5. Shuffle the 1-D sequence $\tilde{G}$ according to the sorted indices in (4) to obtain $\tilde{G}_s$.

6. Obtain a state-space realization of $\tilde{G}$.

7. Find a minimal realization in (6) if it is not already minimal.

8. Approximate this minimal realization by the Hankel model reduction method to obtain an approximate $\hat{G}_r$.

9. Synthesize (decompress) i.e. find the response of the system using $\hat{G}_r$ using $Y_k = CA^{k-1}B$ where the parameters (A, B, C) are for the approximated values, i.e., (A_r, B_r, C_r) . Note that the approximate system will be typically an infinite impulse response (IIR). Truncate the response by any desired window of interest.

10. Reorder the synthesized image impulse response matrix in (9) according to the sorting indices in (4) to obtain the approximate image $\hat{G}$. This is actually the reverse of (4) (i.e. put the blocks back in their natural order as determined in step 2).

11. Display the original image G alongside the synthesized approximated image $\hat{G}_r$ for comparison purposes.

Remark: Why order the samples? To understand this consider the following example. Given the following 1-D finite impulse response (FIR) system

$$F_1(z) = 1 + 2z^{-1} + 3z^{-2} + 4z^{-3} \tag{3.2}$$

which is to be approximated by a first order system, let us say, one option would be, to carry out the approximation of the system as it is and the other would be, to first order the system as :

$$F_2(z) = 4 + 3z^{-1} + 2z^{-2} + 1z^{-3} \tag{3.3}$$

and then carry to out the approximation of the ordered system. The HSV's of $F_1(z)$ are [7.1 3.6 2.4] while that of $F_2(z)$ are [4.4 0.7 0.3]. We thus see, from Nehari's theorem, that a lower bound for the approximation error is (0.7) for the ordered system and for the not ordered case it is (3.6). The conclusion is that the ordered system has smaller approximation error.

3.4.5 Algorithm 6: H_∞ Norm Image Approximation via Least Squares Identification

1. Concatenate the given $M \times N$ image with impulse response matrix G according to some desired manner (row or column-wise) to obtain a 1-D sequence with impulse response matrix $\tilde{G}$.

2. Sub-divide the concatenated image into sub-image frames of specfied length q and size $1 \times q$ hence a total of $\frac{M \times N}{1 \times q}$ 1-D sub-images.

3. Group the elements of each frame in groups of size g, where g is specified. Hence a total of $\frac{q}{g}$ groups in each frame.

4. Compute the spectral norm of each group.

5. Order the $\frac{q}{g}$ groups in a descending order according to their norms. Use a group index i.e., vector $J = (J_1, J_2, ...J_n)$ to keep track of the ordering for later use.

6. Sort the groups of all sub-images frames in a descending order of their norms.

7. Set $k = 0$. Select an initial input sequence $u^0(t)$ as a white random sequence.

8. Compute $y^k(t) = g_n(t) * u^k(t)$, where $g_n(t)$ is the impulse response obtained from the sorted sub-images frames.

9. Obtain $\hat{\theta^k}$ =arg $\min_\theta \sum_{t=0}^{N} |\epsilon_{LS}(t,\theta)|^2$

   with $\epsilon_{LS}(t,\theta) := y^k(t) - \phi^T(t)\theta$

10. Set $u^{k+1}(t) = \frac{\epsilon_{LS}(t,\hat{\theta})^k}{\sum_{t=0}^{N} |\epsilon_{LS}(t,\theta)|}$. Note that the scaling done here is to ensure that the input remains bounded.

11. Set $k = k + 1$ and go to step 8. Note that the iteration is terminated using a pre-selected criterion.

12. Synthesize $\hat{G}_r$ using the approximated parameters, θ^k.
13. Reorder the impulse response vector of the synthesized image in (12) according to the index vector found in (5) .
14. Inverse concatenate the 1-D image response vector in (13) to obtain the approximate 2-D image $\hat{G}$.
15. Display the original image G alongside the synthesized approximated image $\hat{G}$ for comparing purposes.

Note: This algorithm is based on generating a sequence of input-output pairs for the given impulse response obtained from the various groups of the sub-images and identifying the series of reduced order models which, when properly concatenated, would optimally approximate in a H_∞-norm sense, the original transfer function.

3.4.6 Algorithm 7: H_∞ Norm Speech Approximation via Least Squares Identification

The main aim of speech compression is to digitize speech into compact data structure, which when decoded, will provide the best reconstructed vocal quality. In general, most of the research on speech compression has been revolving on performance trade-offs between system intelligibilty(and therefore user acceptance) and transmission data rate [36].

Speech coders can be broadly classified into three groups: waveform coders, vocoders, and hybrid coders. The waveform coders provide high speech quality above 16Kbps but their performance usually falls off rapidly at much lower bit rates. The vocoders(e.g. Linear Prediction Coding-LPC) however, are able to bring the bit rate down to much lower values even as low as 400bps but the speech quality at its best is only fair [37], while the hybrid coders(e.g. Code Excited Linear Prediction -CELP) combines features from vocoders and waveform coders and generally operate better than the two [38].

Although the main focus of this thesis is image compression, we outline here-below a modified version of Algorithm 6 to apply to speech signals.

1. Convert the 8-bit mu-law format sound data to an equivalent ASCII format sound data.
2. Represent the analog sound data in (1) as a sequence of numbers via sampling i.e.

$$x(n) = x_a(nT)$$

   for $-\infty < n < \infty$, where n is integer and T is the sampling period.
3. Sub-divide the sampled sound data into sub-audio frames each of size q.
4. Group the elements of the frame in groups of size g, where 'g' is specified.
5. Compute the short-time energy at sample n i.e for each group, which is the sum of the squares of the N samples $n - N + 1$ through n. A simple

definition of the short time energy is

$$E_n = \sum_{m=n-N+1}^{n} x^2(m)$$

Figure 3.3 depicts the computation of the short time energy sequence. Note that the window literally slides along the sequence of squared values, selecting the interval to be involved in the computation. This windowing process represents the grouping operation mentioned in step 3 of Algorithm 6.

6. Find the ordering from largest to the smallest short-time energies(norms) of these sub-groups.

7. Shuffle the speech sequence $\tilde{G}$ according to the sorted indices in (6) to obtain $\tilde{G}_s$

8. Set $k = 0$. Select an initial input sequence $u^0(t)$ as a white random sequence.

9. Compute $y^k(t) = g_n * u^k(t)$, where $g_n(t)$ is the impulse response obtained from the sorted sub-images frames.

10. Obtain $\hat{\theta}^k$ =arg $min_\theta \sum_{t=0}^{N} |\epsilon_{LS}(t,\theta)|^2$

    with $\epsilon_{LS}(t,\theta) := y^k(t) - \phi^T(t)\theta$

11. Set $u^{k+1}(t) = \frac{\epsilon_{LS}(t,\hat{\theta})^k}{\sum_{t=0}^{N} |\epsilon_{LS}(t,\theta)|}$

12. Set $k = k + 1$ and go to step 9. Note that the iteration is terminated using a pre-selected criterion.

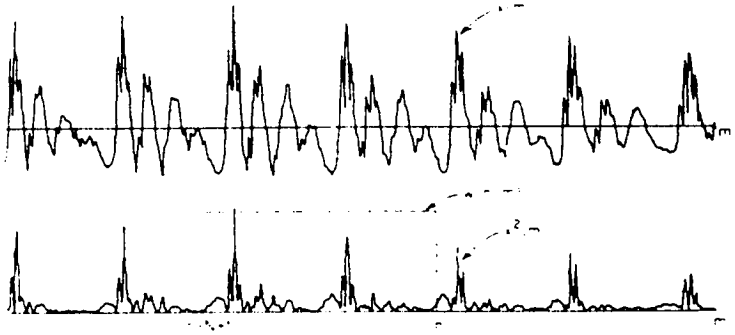


Figure 3.3 : Illustration of computation of speech short time energy

13. Synthesize (decompress) $\hat{G}_r$ using the approximated parameters.

14. Play the original and approximated speech signal . Also display their respective frequency response plots for comparison purposes.

Note: The statistical properties of a speech signal changes with time i.e. it is not stationary. The underlying assumption in most speech processing schemes is that the properties of the speech signal change relatively slowly with time. This assumption led to a variety of "short-time" processing methods in which short segments of the speech signal are isolated and are processed as if they were short segments from a sustained sound with fixed properties as if they were stationary. This is repeated periodically as often as desired. These short segments are sometimes called analysis frames. The length of analysis frames used in practice are such thaat the one-way(round trip) delay does not exceed 50 milliseconds. The result of the processing of each frame may be either a single number or a set of numbers. Therefore such processing produces a new time- dependent sequence which can serve as a representation of the speech signal. The short-time amplitude of a speech signal will thus provide a convenient representation of the speech amplitude variations [2].

Chapter 4

Simulation Results and Performance Evaluation

4.1 Introduction

One way of validating a developed algorithm is by implementing it practically. This often leads to certain revealing or intriguing insights into the performance of the algorithm and may further provide the necessary impetus required for taking care of any additional problems that may arise in its real-world application.

Numerous experiments were conducted to evaluate the performance of the algorithms developed in the previous chapter. The simulations were carried

out using the Matlab package with its toolboxes on a SUN SPARC 10 station as the platform.

Several real-life images were used to evaluate the algorithms. Some illustrative examples are presented here. A diskette containing a collection of test images is provided on the back flap cover of this thesis. Also included in the diskette is the speech signal data used.

This chapter is divided as follows: The two performance measures for image compression, namely the subjective (qualitative) and objective (quantitative) image fidelity parameters are presented in sub-sections 2.1 and 2.2 respectively. The performance measure for speech compression is given in section 3. Image data compression using algorithms 1-4 are addresed in sub-section 4.1. Sub-section 4.2 covers image data compression using algorithms 1-5. H_∞ image approximation is the focus of sub-section 4.3, some cross-reference is also made in this sub-section to the DCT-based compression whilst sub-section 4.4 covers speech compression.

4.2 Image Fidelity Criteria

Because information of interest may be lost during data compression, a repeatable or reproducible means of quantifying the nature and extent of information loss is highly desirable [1]. Image fidelity criteria are also useful for measuring

image quality and also for rating the performance of a processing technique or a vision system [4]. Two general classes of criteria are used as the basis for an assessment : (1)objective(i.e quantitative) fidelity criteria and (2) subjective (qualitative) fidelity criteria [1, 4].

4.2.1 Objective Fidelity Criteria

When the level of information loss can be expressed as a function of the original or input image and the compressed or subsequently decompressed output image, it is said to be based on an objective fidelity criterion. A good example is the root-mean-square-error between an input and output image [1]. Another popular type of criterion is called the mean-square criterion. It is based on minimizing the average of sum of squares of the error between two images [4]. Let $U(m,n)$ represent an $M \times N$ input image and let $\hat{U}(m,n)$ denote an estimate or approximation of $U(m,n)$ that results from compression or decompression. The following quantitative measures are defined [1, 4]:

1. Error:

$$e(m,n) = \hat{U}(m,n) - U(m,n) \tag{4.1}$$

2. Total error:

$$te = \sum_{m=1}^{M} \sum_{n=1}^{N} [\hat{U}(m,n) - U(m,n)] \tag{4.2}$$

3. Root-mean-square error:

$$rms = \left[\frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}[\hat{U}(m,n) - U(m,n)]^2\right]^{1/2} \tag{4.3}$$

4. Average least squares error:

$$alse = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}|\hat{U}(m,n) - U(m,n)|^2 \tag{4.4}$$

5. Signal-to-noise ratio:

$$snr = 10\log\frac{\sigma^2}{\sigma_e^2} \tag{4.5}$$

σ^2 is the variance of the desired(original)image, while σ_e^2 is the variance of the approximated/estimated image.

6. Mean square signal-noise-ratio (of compressed-decompressed image):

$$mssnr = \frac{\sum_{m=1}^{M}\sum_{n=1}^{N}\hat{U}(m,n)^2}{\sum_{m=1}^{M}\sum_{n=1}^{N}[\hat{U}(m,n) - U(m,n)]^2} \tag{4.6}$$

and the root mean square SNR is just the square-root of it.

7. Mean

$$\mu = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}U(m,n) \tag{4.7}$$

8. Variance

$$\sigma^2 = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}[U(m,n)^2 - \mu^2] \tag{4.8}$$

9. Correlation Coefficient

$$r = \frac{\sum_{m=1}^{M}\sum_{n=1}^{N}U(m,n)\hat{U}(m,n)}{\left(\sum_{m=1}^{M}\sum_{n=1}^{N}U^2(m,n)\right)^{1/2}\left(\sum_{m=1}^{M}\sum_{n=1}^{N}\hat{U}^2(m,n)\right)^{1/2}} \tag{4.9}$$

10. Energy

From Parseval's theorem, we have:

$$\frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}|U(m,n)|^2 = \frac{1}{4\pi^2}\int_{-\pi}^{\pi}\int_{-\pi}^{\pi}|U(w_1,w_2)|^2 dw_1 dw_2 \qquad (4.10)$$

4.2.2 Subjective Fidelity Criteria

Although objective fidelity criteria offer a simple and convenient mechanism for evaluating information loss, most decompressed images are ultimately viewed by human beings. Consequently, measuring image quality by subjective evaluations of a human observer is often more appropriate [1]. The subjective criteria use rating scales such as goodness scales(Table 4.1) and impairment scales(Table 4.2) [4]. This can be accomplished by showing a typical decompressed image to an approximate cross section of viewers and averaging their evaluations. The evaluations may be made using an absolute rating scale or by means of side-by-side comparison of $U(m,n)$ and $\hat{U}(m,n)$. In either case the observations are said to be based on subjective fidelity criteria [1].

4.3 Speech Signal Quality

Some commonly used interpretations for speech quality or fidelity are signal-to-noise ratio, intelligibility, communicability and subjective preference or scaling

Table 4.1. Image goodness scales

(i)overall goodness scale(side-to-side with original)	(ii)group goodness scale(set of approximate images)
Excellent (extremely high quality) (5)	Best (7)
Good (high quality) (4)	Well above average (6)
Fair (acceptable quality) (3)	Slighty above average (5)
Poor (poor but watchable) (2)	Average (4)
Unsatisfactory (so bad) (1)	Slightly below average (3)
	Well below average (2)
	Worst (1)

Table 4.2. Image impairment scales (based on degradation levels)

Not noticeable (1)
Just noticeable (2)
Definetely noticeable but only slightly (3)
Impairment not objectionable (4)
Somewhat objectionable (5)
Definetely objectionable (6)
Objectionable (7)

Table 4.3. Speech Signal Quality

Rating	Speech Quality	Level of distortion
5	Excellent	Imperceptible
4	Good	Just Perceptible, but not annoying
3	Fair	Perceptible and slightly annoying
2	Poor	Annoying, but not objectionable
1	Unsatisfactory	Very annoying and objectionable

evaluations. Although most of these are valuable, given the proper applications, there are a wide range of situations in which the fidelity measures are either misleading or even completely inappropriate [39].

Obviously, any meaningful definition of speech quality must be based on human responses and perception. Objective quality measures, on the other hand, attempt to measure those physical characteristics of the speech signal that are correlated with factors that determine speech quality. Since the success of an objective measure is generally evaluated by its ability to predict some subjective quality assessment, the performance of an objective measure cannot be disassociated from the subjective quality measure it estimates [39].

Since listeners are required to judge the test speech in terms of the reference speech, listeners may be forced into a smaller perceptual descriptor than might be desired [39].

The most widely used direct method of subjective quality evaluation is the category method, which results in a mean optimum score(MOS). In this method, listeners rate the speech under test on the five-point scale in Table 4.3 in which the listeners' impressions are assigned a numerical value. Though five categories may seem too few, Miller studies [40] of the limits of human information processing indicate that as few as five but no more than nine categories should be used. The measured quality of the speech under test is equal to the average

value of the scores received from listeners. If desired, data from all listeners can be adjusted to zero mean, and unit variance (i.e. normalized) prior to averaging to remove any individual bias.

4.4 Results and Evaluation

4.4.1 Case I

In this example, Algorithms 2 through 5 were implemented to analyse their performances . One of the test images used is a 500 × 500 indexed image titled 'Mandrill'.

The image is then divided into blocks each of size 25 × 25 resulting in a total of 400 blocks (step 1 of the algorithm). The 500 × 500 mandrill image is then mapped into a 25 × 10000 mandrill image (step 2). The subsequent steps of these algorithms(i.e. 2-4) are then carried out sequentially. Note that in step (7), to obtain the state -space realization, the 25 × 10000 mandrill image was partitioned into segments of sizes 25 × 125 which gives a total of 80 segments. Each of these segments is then realized separately with each segment state-space parameters having sizes 100 × 100, 100 × 25, 25 × 100 , and 25 × 25 respectively.

The question now is how to determine the order of the parameters for the

approximated image segments. Extensive simulation studies were carried out on the mandrill image segments with the order of the approximate system being determined by eliminating states whose singular values are less than one-tenth of the maximum singular value. This resulted in a reduced order of 15 for which intelligible image reproduction was achieved. Further test trials with orders less than 15 were carried out and revealed that a minimum order of 10 can be used for meaningful image reproduction. Based on this, the order of the approximated image parameters was set to 10. Thus the approximated parameters A, B, C, D of each image segment were of sizes $10 \times 10, 10 \times 25, 25 \times 10$ and 25×25 respectively.

The quantitative image fidelity parameters obtained by applying algorithms 1 through 4 to the image mandrill are shown in Table 4.4. Four approximated images to the original image (Figure 4.1) are obtained viz: sorted balanced model reduced image(SBMRI) in Figure 4.2, sorted Hankel model reduced image(SHMRI) in Figure 4.3, non-sorted balanced model reduced image(NSBMRI) in Figure 4.4 and non-sorted Hankel model reduced image(NSHMRI) in Figure 4.5.

The compression ratio for each segment is $C = \frac{100}{10} = 10$ while the redundancy for each segment is $R = 0.9$

thus the reduction rate in percent for the whole image $= 100 - 80 \times 0.1 = 92\%$.

Table 4.4. Image Fidelity Quantitative Parameters.

Fidelity measures \ Image type	Original	Sorted Balanced (SBMR)	Sorted Hankel (SHRM)	Non-sorted Balanced (NSBRM)	Non-sorted Hankel (NSHRM)
Total error		-19.4716	-466.9163	-43.7609	-507.321
Root mean square error		0.0955	0.0900	0.0939	0.0886
Average least squares error		0.0091	0.0081	0.0088	0.0079
Signal-to-noise-ratio		5.8597	6.0094	6.0213	6.1856
Mean square signal-to-noise ratio		1.2134e+02	0.2026	24.0246	0.1748
Root mean square signal-to-noise ratio		11.0152	0.4541	4.9015	0.4181
Mean	0.3859	0.3858	0.3840	0.3857	0.3839
Standard Deviation	0.1876	0.1875	0.1799	0.1877	0.1806
Correlation		0.9752	0.9778	0.9761	0.9785
Energy	4.6025e+04	4.6004e+04	4.4958e+04	4.6007e+04	4.4999e+04

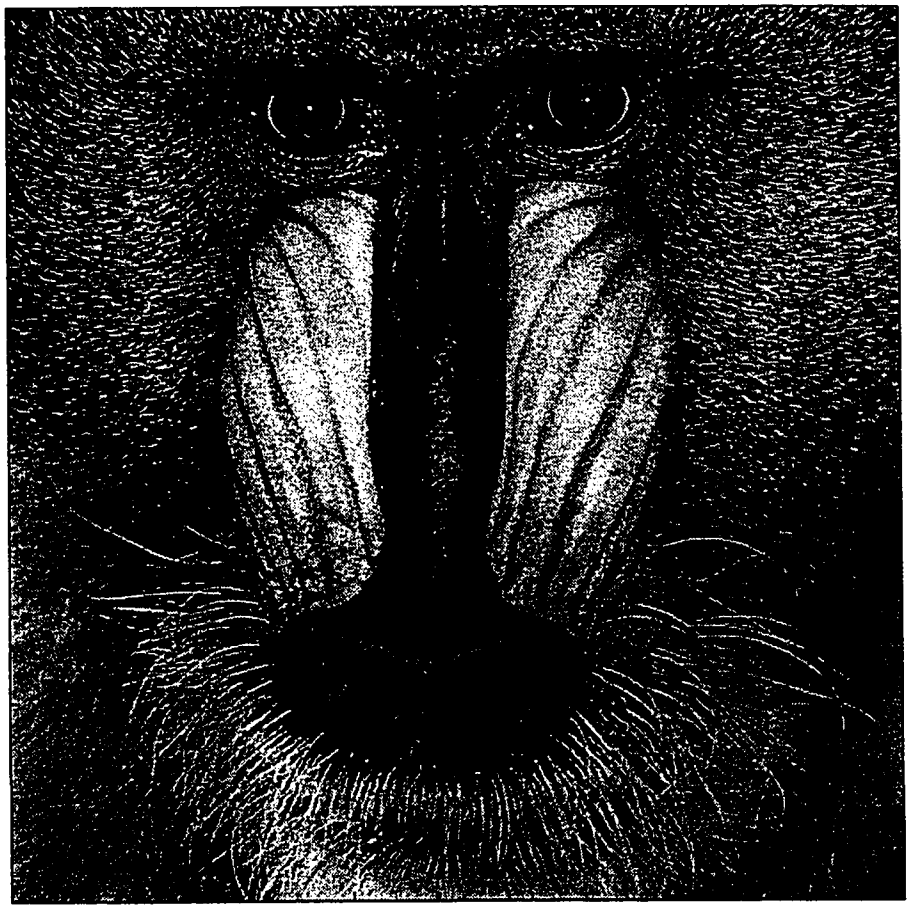


Figure 4.1 : Original 'Mandrill' image.

Figure 4.2 : SBMR image. RMSE=0.0955. SNR=5.8597dB

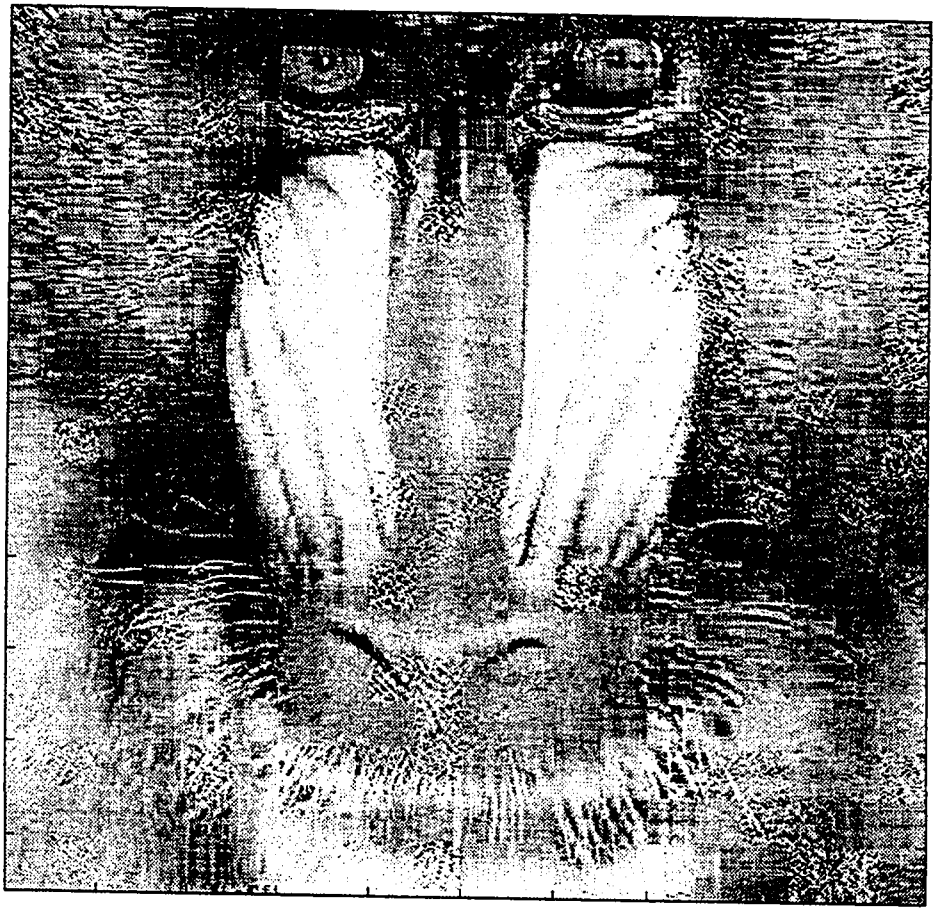


Figure 4.3 : SHMR image. RMSE=0.0900. SNR=6.0094dB

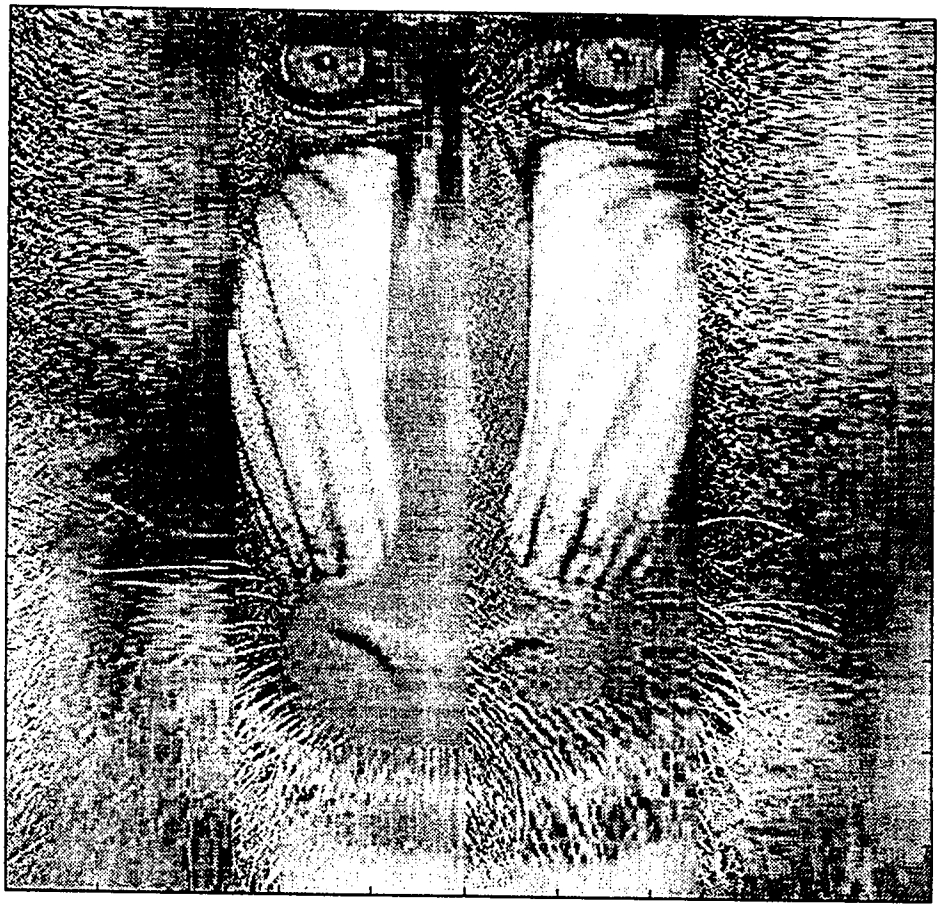


Figure 4.4 : NSBMR image. RMSE=0.0939. SNR=6.0213dB

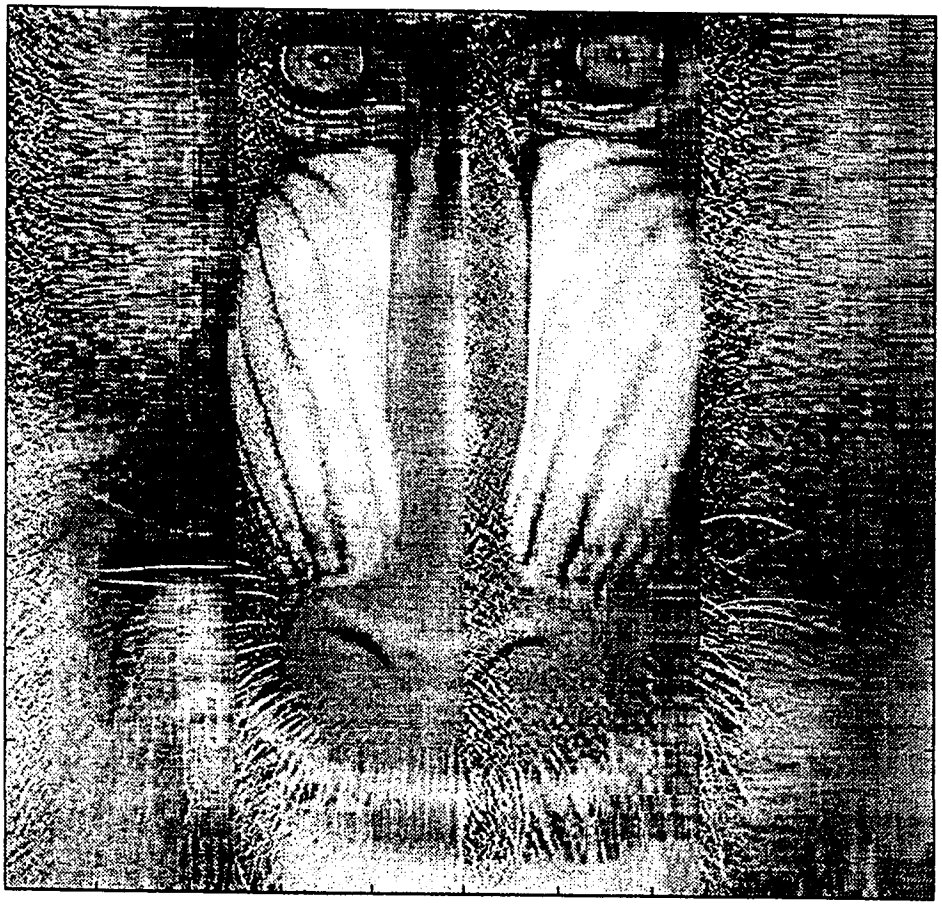


Figure 4.5 : NSHMR image. RMSE=0.0886. SNR=6.1856dB

(Note that 80 is the total number of segments).

From the SNR and RMSE(which are two very important performance measures for quantitative image fidelity) values in Table 4.4 of the approximated images, it can be seen that the performance of the SHMRI is better than that of the SBMRI. The same is also true of the NSHMRI to the NSBMRI. This is not unexpected, since we know from system theory that the error norm of the Hankel norm approximation scheme is tighter than that of the Balanced model reduction scheme. In addition, the sorted versions results in a smaller error than their unsorted counterparts as seen in Table 4.4

The SHMRI and NSHMRI are more closer to the original image as indicated by their correlation coefficients. However the mean and variance of the SBMRI and NSBMRI are higher indicating that their average brightness and measure of contrast is closer to the original image than the rest.

4.4.2 Case II

In this example, Algorithms 1 through 5 were implemented to analyse and compare the performance of the DCT based compression algorithm(i.e. algorithm 1) with the Balanced and Hankel model reduction based compression algorithms(i.e. algorithms 2-5) discussed in Case I. The test image used is a 512 × 512 image titled 'Mandrill'. Processing this image at 8 bits/pixel

amounts to an original image size of 2MB (Megabyte).

The sub-image block size was taken as 32 × 32 giving a total of 256 blocks each of 1024 pels. The four algorithms of case I and the DCT algorithm outlined in section 3.1 were each applied individually to the original input image. This was carried out for different values of compression ratios up-to a ratio at which a noticeable level of degradation starts to appear. Five approximated images to the original (see Figure 4.1) were obtained for each algorithm. Figures 4.6 through 4.10 shows the approximated images (sorted Balanced model reduced image-SBRI; sorted Hankel model reduced image-SHRI; non-sorted Balanced model reduced image-NSBRI; non-sorted Hankel model reduced image-NSHRI; and DCT compressed image) for a compression ratio of 8. Qualitative and quantitative image fidelity parameters were measured and recorded. Tables 4.5 through 4.8 give all the pertinent quantitative image fidelity parameters whilst Tables 4.9 through 4.11 give the qualitative (subjective) image fidelity parameters based on the standard tables (Tables 4.1 & 4.2) given in section 4.2. Figures 4.11 through 4.14 show the plot of the quantitative parameters. From Table 4.5 and Figure 4.12, we see that the DCT-based compression technique is almost independent or insensitive to the reconstruction error and signal-to-noise ratio. The DCT and SBR compressed images posseses almost the same energy for the different compression ratios indicated in Figure 4.14.

SBRI. RMSE=0.1084, SNR=5.4201, Ratio=8

Figure 4.6 : SBMR image for block size 32 X 32 .

SHRI. RMSE=0.1028, SNR=5.4395, Ratio=8

Figure 4.7 : SHMR image for block size 32 X 32 .

NSBRI. RMSE=0.1083, SNR=5.4262, Ratio=8

Figure 4.8 : NSBMR image for block size 32 X 32 .

NSHRI. RMSE=0.1021, SNR=5.5177, Ratio=8

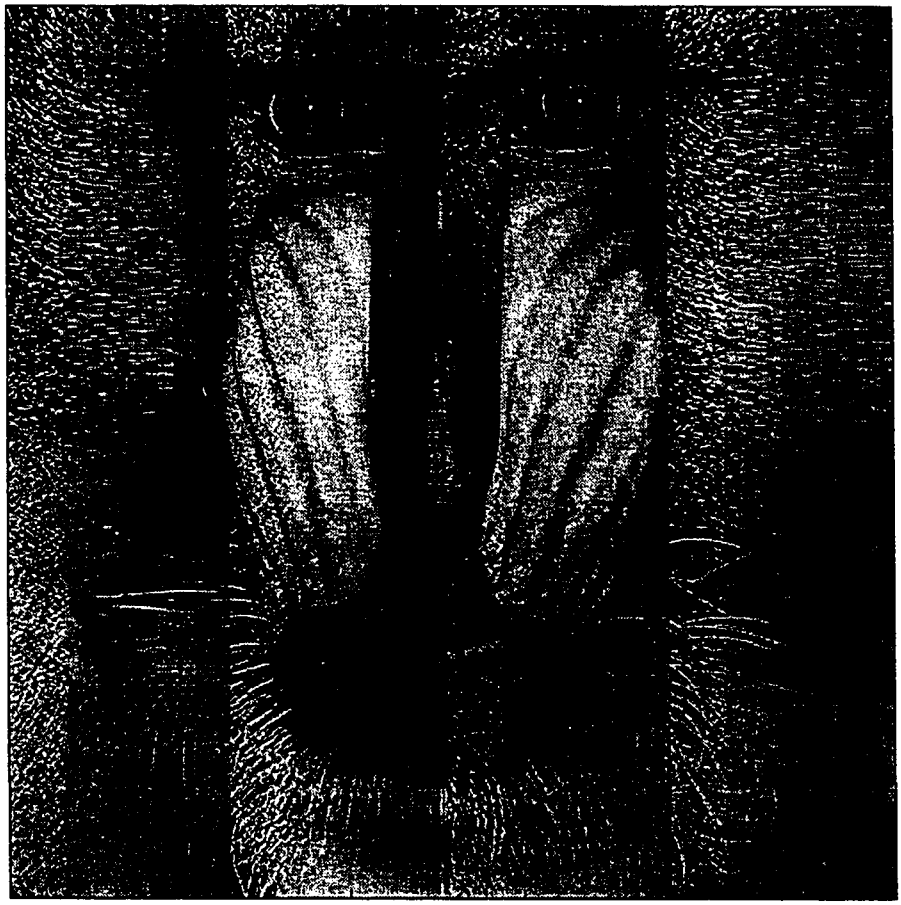


Figure 4.9 : NSHMR image for block size 32 X 32 .

DCTI. RMSE=0.1097, SNR=3.8070, Ratio=8

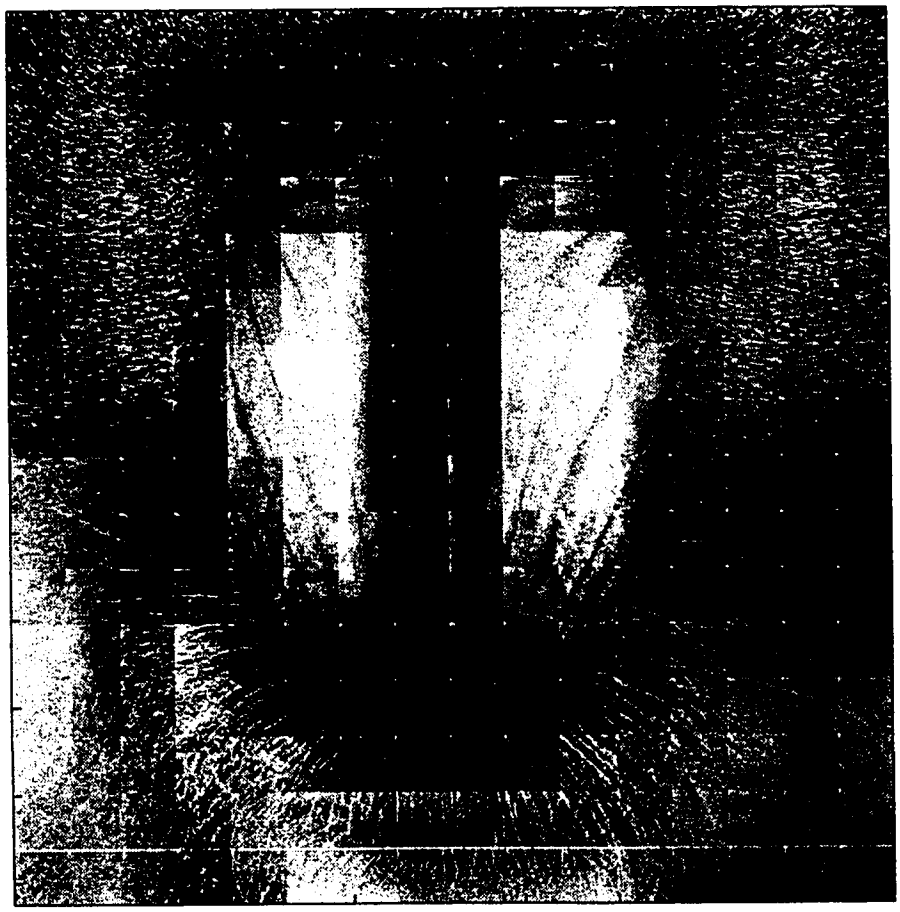


Figure 4.10 : DCT image for block size 32 X 32 .

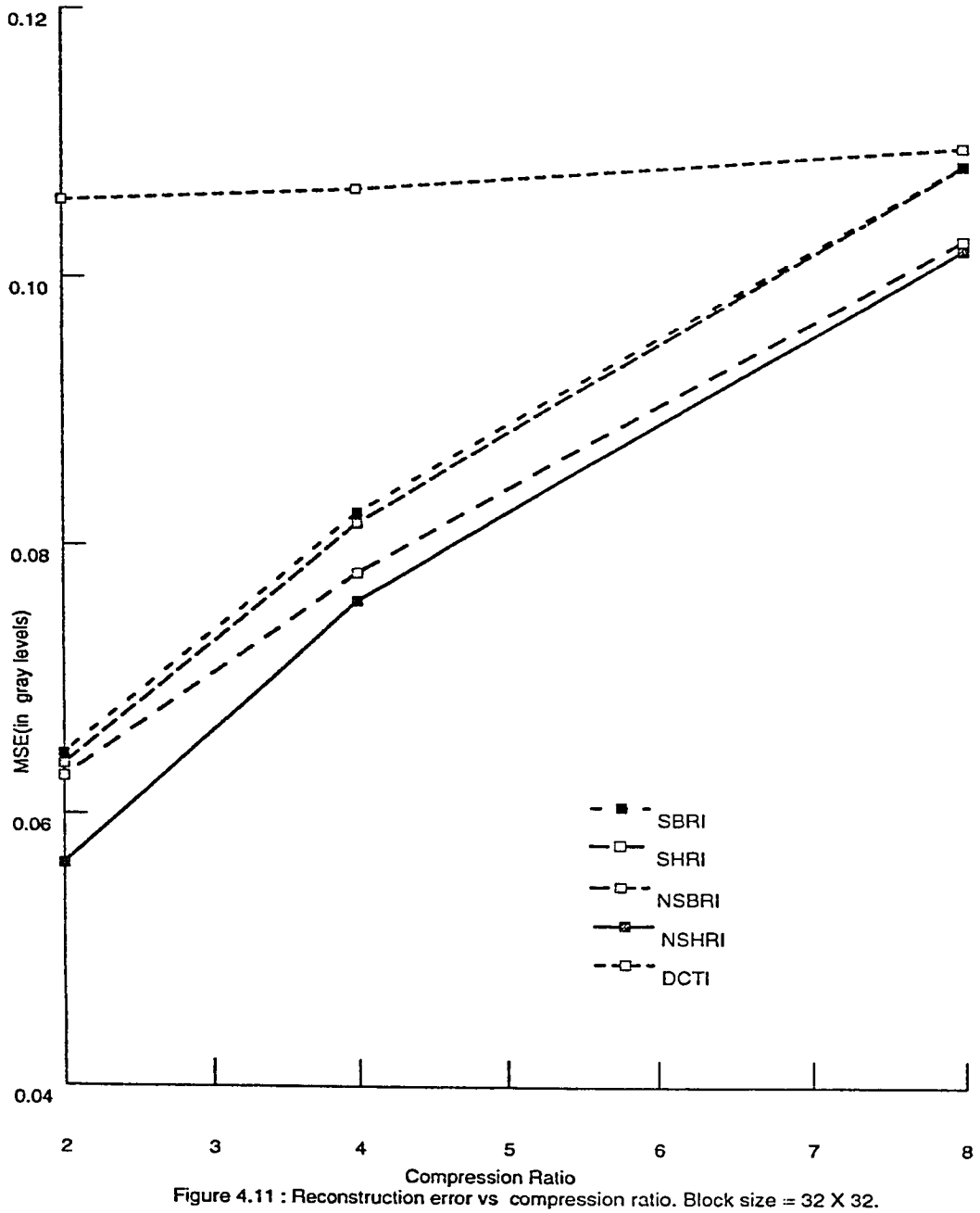


Figure 4.11 : Reconstruction error vs compression ratio. Block size = 32 X 32.

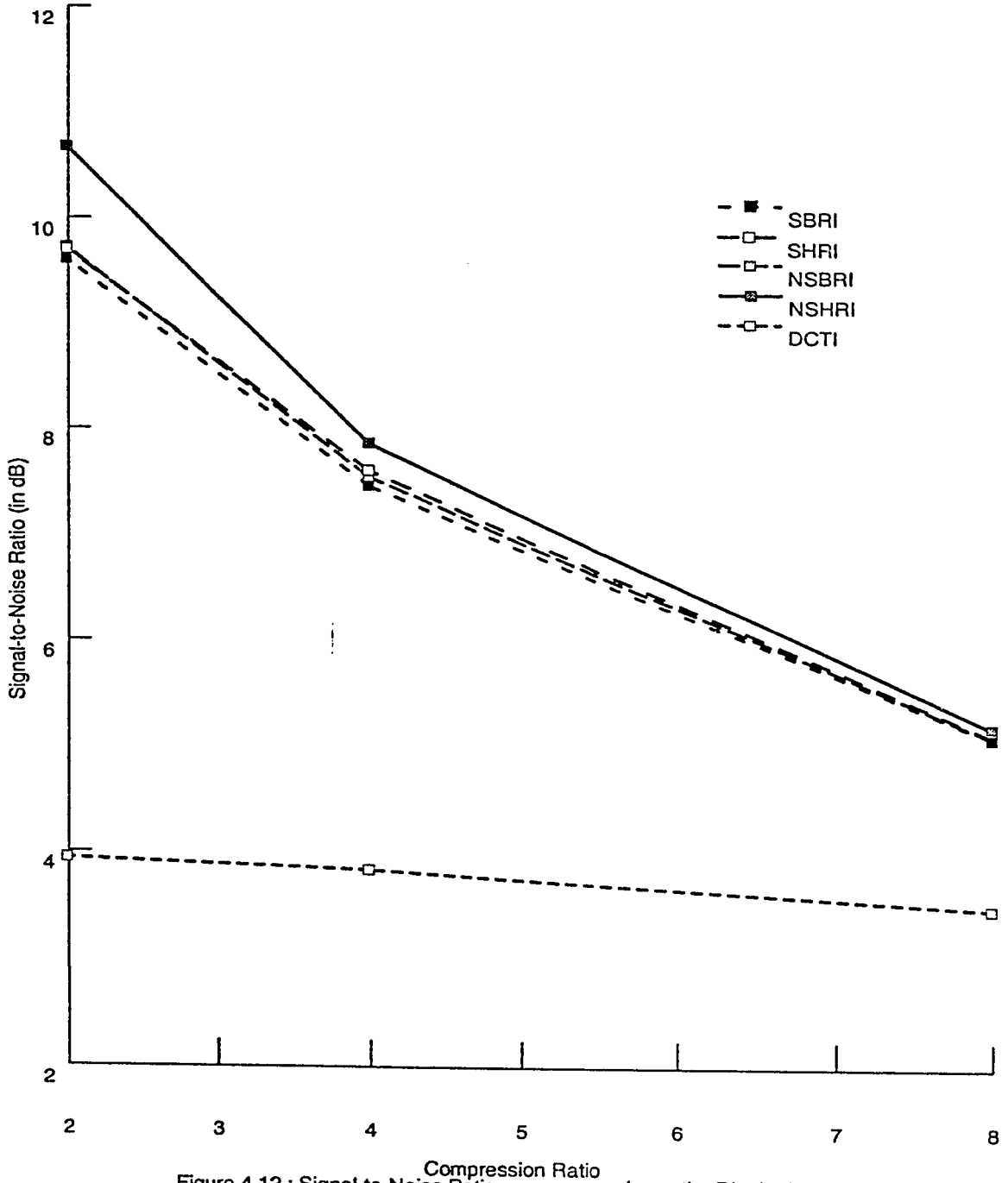


Figure 4.12 : Signal-to-Noise Ratio vs compression ratio. Block size = 32 X 32.

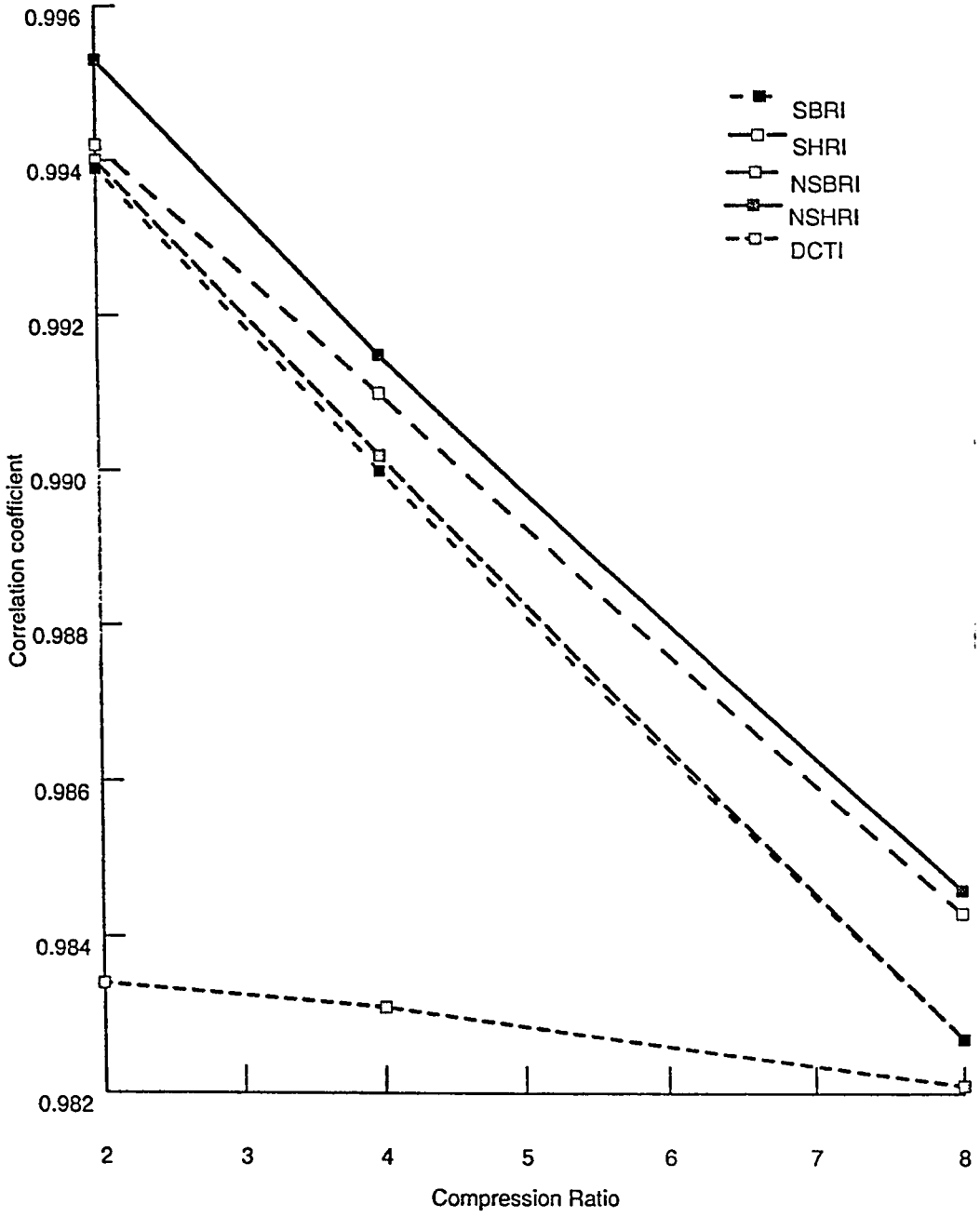


Figure 4.13 : Correlation coefficient vs compression ratio. Block size = 32 X 32.

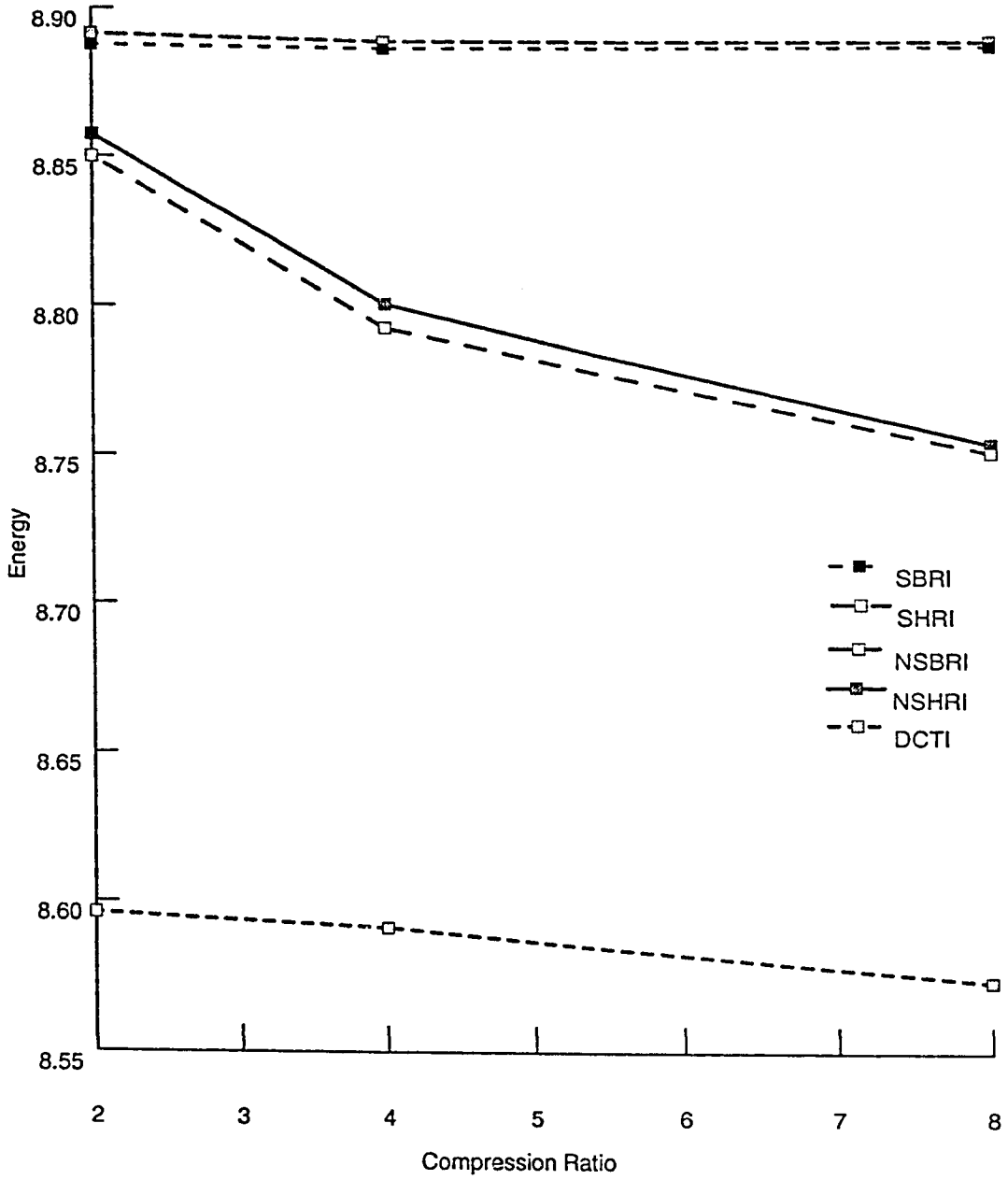


Figure 4.14 : Energy vs compression ratio. Block size = 32 X 32.

Table 4.5 : Mean-Square Error (MSE) in gray levels.

Ratio	Image size	SBRI	SHRI	NSBRI	NSHRI	DCTI
2	1MB	0.0645	0.0628	0.0637	0.0563	0.1057
4	512KB	0.0825	0.0780	0.0817	0.0759	0.1066
8	256KB	0.1084	0.1028	0.1083	0.1021	0.1097

Block size : 32 X 32.

Table 4.6 : Signal-to-Noise Ratio (in decibels).

Ratio	Image size	SBRI	SHRI	NSBRI	NSHRI	DCTI
2	1MB	9.9224	10.0211	10.0387	10.9902	4.2544
4	512KB	7.7844	7.9298	7.8720	8.1893	4.1579
8	256KB	5.4201	5.4395	5.4262	5.5177	3.8070

Block size :32 X32.

Table 4.7 : Correlation Coefficient

Ratio	Image size	SBRI	SHRI	NSBRI	NSHRI	DCTI
2	1MB	0.9939	0.9942	0.9940	0.9953	0.9834
4	512KB	0.9900	0.9910	0.9902	0.9915	0.9831
8	256KB	0.9827	0.9843	0.9827	0.9846	0.9821

Block size : 32 X 32.

Table 4.8 : Energy (e+04)

Ratio	Image size	SBRI	SHRI	NSBRI	NSHRI	DCTI
2	1MB	8.8876	8.8499	8.8914	8.8573	8.5962
4	512KB	8.8867	8.7929	8.8891	8.8009	8.5914
8	256KB	8.8881	8.7515	8.8898	8.7544	8.5737

Block size :32 X32.

Table 4.9 : Qualitative (subjective) measures.

Parameters	SBRI	SHRI	NSBRI	NSHRI	DCTI
Overall goodness scale	5	5	5	5	2
Group goodness scale	6	6	7	7	3
Impairment scale	2	2	1	1	6

Block size: 32 X 32. Ratio =2 (Image size = 1MB).

Table 4.10 : Qualitative (subjective) measures

Parameters	SBRI	SHRI	NSBRI	NSHRI	DCTI
Overall goodness scale	4	4	4	4	3
Group goodness scale	5	5	5	5	3
Impairment scale	2	2	2	2	6

Block size: 32 X 32. Ratio =4(Image size = 512KB).

Table 4.11 : Qualitative (subjective) measures

Parameters	SBRI	SHRI	NSBRI	NSHRI	DCTI
Overall goodness scale	3	3	2	2	3
Group goodness scale	4	5	3	3	3
Impairment scale	3	3	4	4	6

Block size: 32 X 32. Ratio =8(Image size = 256KB).

The simulation was also repeated for a sub-image block size of 16 × 16(giving a total of 1024 blocks each of 256 pels). Tables 4.12 through 4.15 gives pertinent quantitative image fidelity parameters whilst Tables 4.16 through 4.18 gives the qualitative (subjective) image fidelity parameters again based on the standard tables. Figures 4.15 through 4.19 shows the approximated images for a compression ratiot of 2, whilst Figures 4.20 through 4.24 shows that for a compression ratio of 4 and figures 4.25 through 4.29 shows for a compression ratio of 8.

It was necessary to investigate the DCT algorithm using two different block sizes due to the fact that the DCT compression algorithm can be implemented for block sizes ranging from 8 × 8 to 64 × 64 . The most popular block sizes for DCT are 8 × 8 and 16 × 16. However the optimal sub-image block size for DCT is 16 × 16 [1,2]. For the DCT-based algorithm, the value of the one-step correlation is also a factor to consider. For $\rho \leq 0.9$, 16 × 16 size is adequate. A rule of thumb is to pick the sub-image size N such that the $\rho^N \ll 1$, say 0.2 [4].

For all the different test cases it can be seen that the Hankel approximation based schemes have the lowest mean square error implying the smallest reconstruction errors followed by the balanced based schemes whilst the DCT based approach has the largest errors.

Table 4.12 Mean-Square Error (MSE) in gray levels.

Ratio	Image size	SBRI	SHRI	NSBRI	NSHRI	DCTI
2	1MB	0.0650	0.0584	0.0636	0.0549	0.0961
4	512KB	0.0850	0.0800	0.0836	0.0781	0.0971
8	256KB	0.11142	0.1099	0.1128	0.1082	0.1008

Block size : 16 X 16.

Table 4.13 Signal-to-Noise Ratio (in decibels).

Ratio	Image size	SBRI	SHRI	NSBRI	NSHRI	DCTI
2	1MB	9.8602	10.6698	10.0441	11.2161	5.3645
4	512KB	7.5207	7.7288	7.6622	7.9637	5.2427
8	256KB	4.9763	4.8919	5.0918	5.0886	4.8146

Block size :16 X16.

Table 4.14 :Correlation Coefficient

Ratio	Image size	SBRI	SHRI	NSBRI	NSHRI	DCTI
2	1MB	0.9938	0.9950	0.9940	0.9953	0.9863
4	512KB	0.9893	0.9905	0.9897	0.9910	0.9860
8	256KB	0.9808	0.9821	0.9812	0.9826	0.9849

Block size : 16 X 16.

Table 4.15 : Energy (e+04)

Ratio	Image size	SBRI	SHRI	NSBRI	NSHRI	DCTI
2	1MB	8.8869	8.8561	8.8887	8.8597	8.6474
4	512KB	8.8848	8.7857	8.8844	8.7925	8.6421
8	256KB	8.8875	8.7258	8.8859	8.7325	8.6228

Block size :16 X16.

Table 4.16 Qualitative (subjective) measures.

Parameters	SBRI	SHRI	NSBRI	NSHRI	DCTI
Overall goodness scale	4	4	4	4	3
Group goodness scale	7	7	7	7	7
Impairment scale	1	1	1	1	6

Block size: 16 X 16. Ratio =2 (Image size = 1MB).

Table 4.17 : Qualitative (subjective) measures

Parameters	SBRI	SHRI	NSBRI	NSHRI	DCTI
Overall goodness scale	4	4	4	4	3
Group goodness scale	7	7	7	7	4
Impairment scale	1	1	1	1	5

Block size: 16 X 16. Ratio =4(Image size = 512KB).

Table 4.18 : Qualitative (subjective) measures

Parameters	SBRI	SHRI	NSBRI	NSHRI	DCTI
Overall goodness scale	3	3	2	2	2
Group goodness scale	6	7	5	4	3
Impairment scale	3	3	4	5	5

Block size: 16 X 16. Ratio =8(Image size = 256KB).

SBRI. RMSE=0.0650, SNR=9.8602, Ratio=2, Block size=16 X16

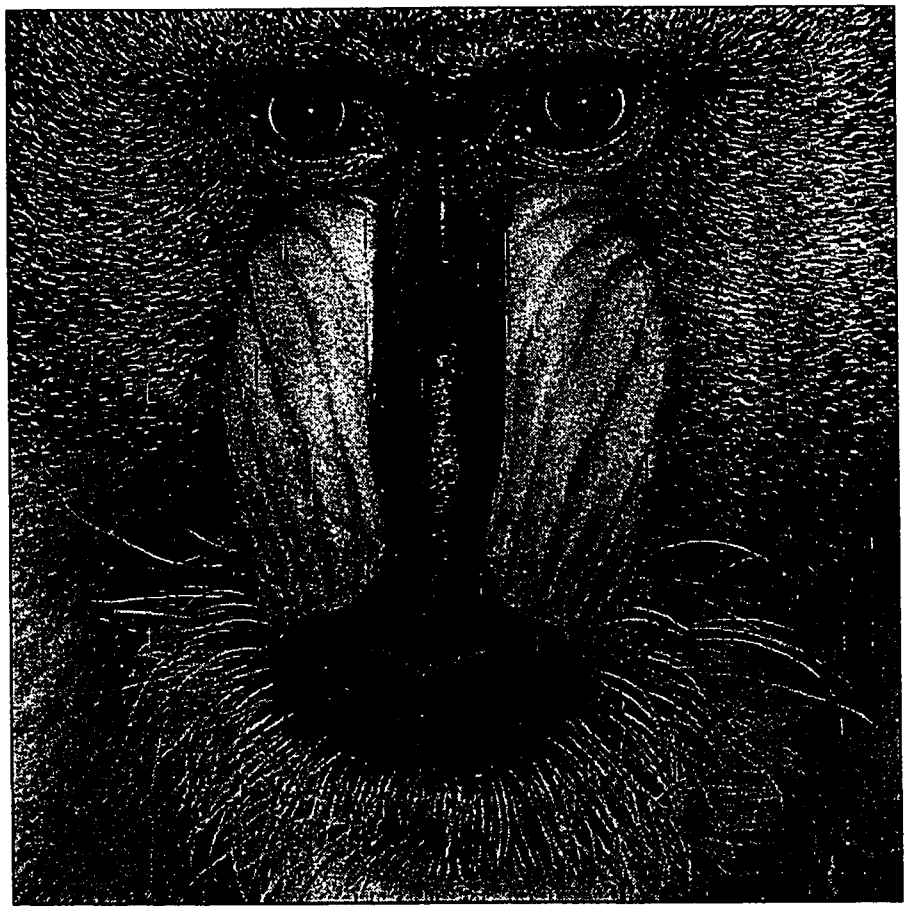


Figure 4.15 : SBMR image for block size 16 X 16 .

SHRI. RMSE=0.0584, SNR=10.6698, Ratio=2, Block size=16 X 16

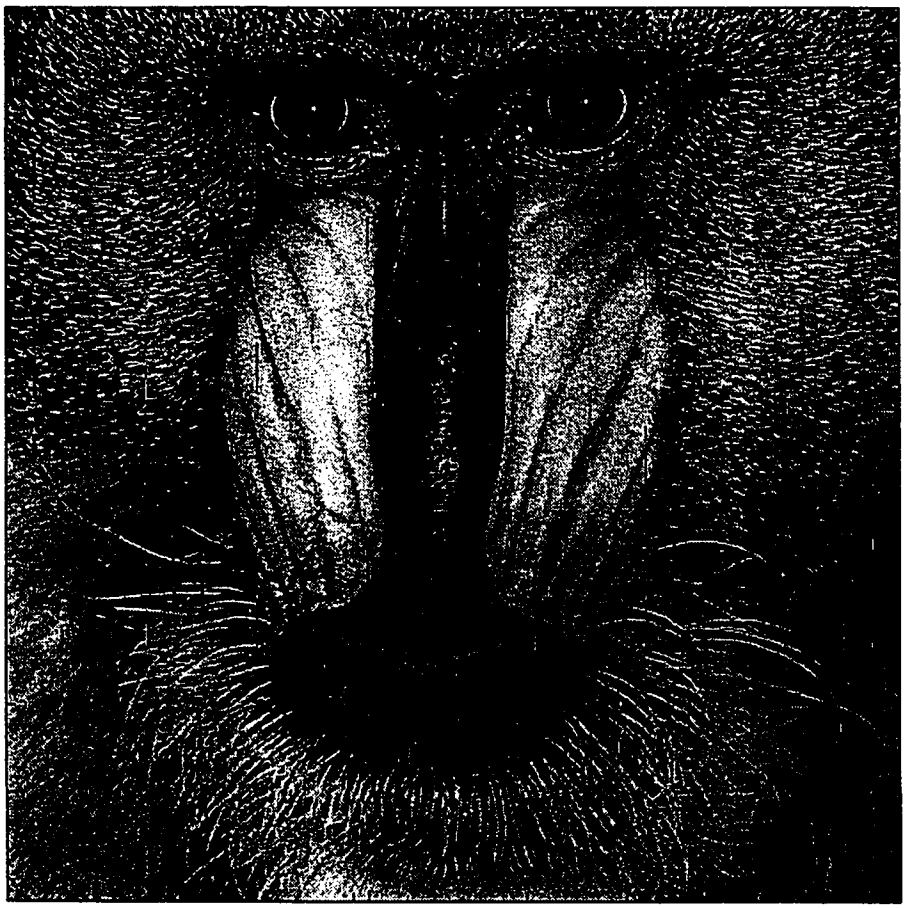


Figure 4.16 : SHMR image for block size 16 X 16 .

NSBRI. RMSE=0.0636, SNR=10.0441, Ratio=2, Block size=16 X 16

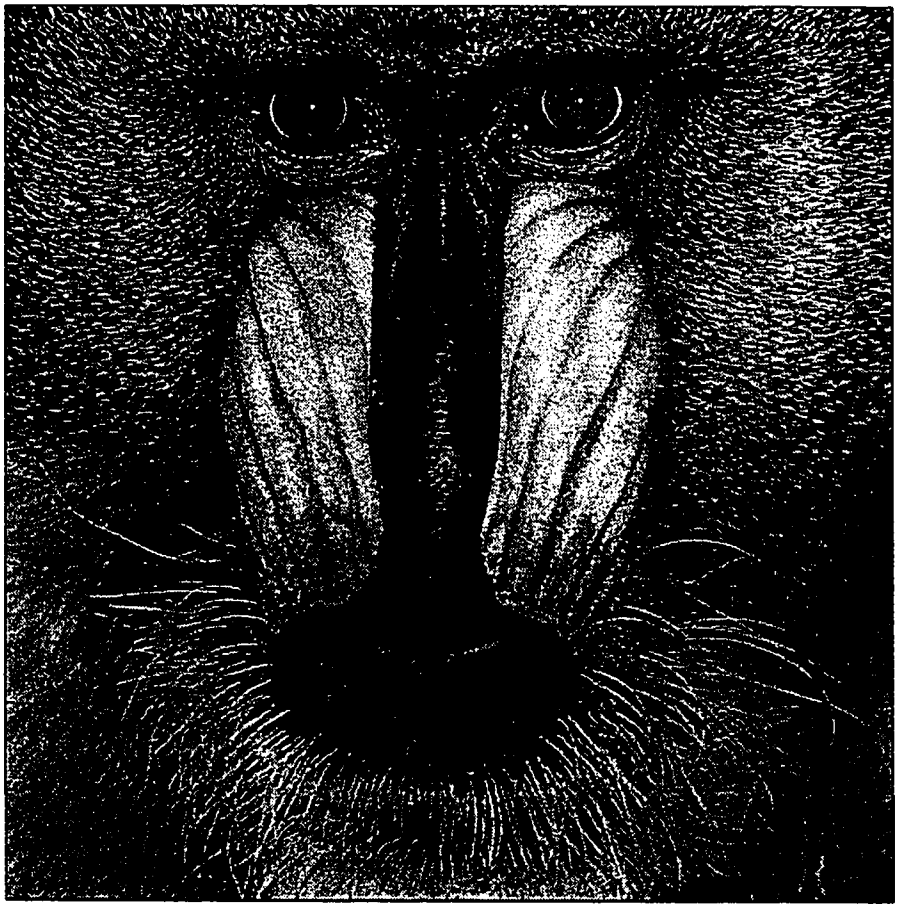


Figure 4.17 : NSBMR image for block size 16 X 16 .

NSHRI. RMSE=0.0549, SNR=11.2161, Ratio=2, Block size=16 X 16

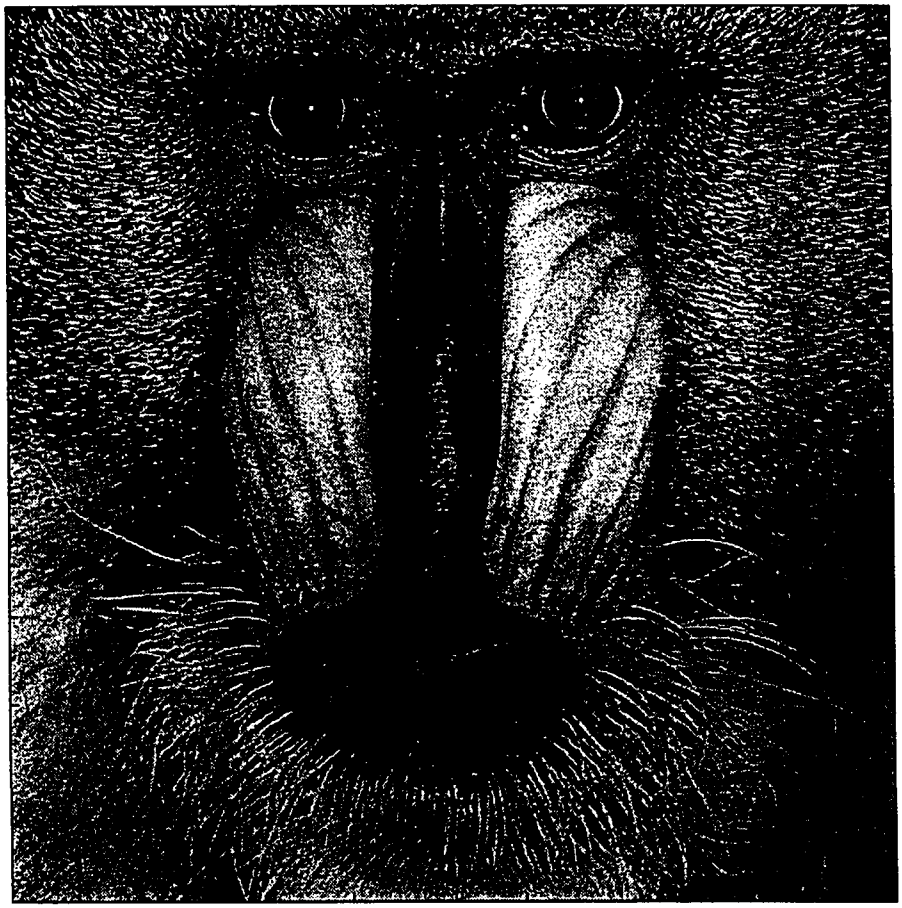


Figure 4.18 : NSHMR image for block size 16 X 16 .

DCTI. RMSE=0.0961, SNR=5.3645, Ratio=2, Block size = 16 X 16

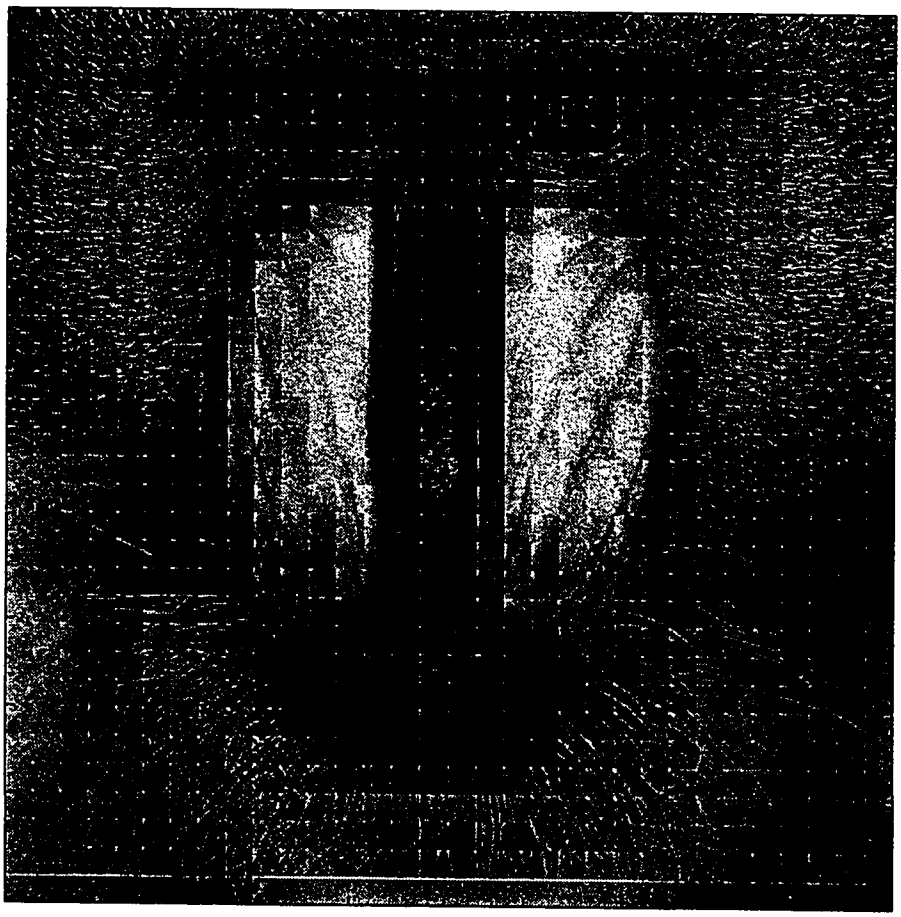


Figure 4.19 : DCT image for block size 16 X 16 .

SBRI. RMSE=0.0850, SNR=7.5207, Ratio=4, Block size=16X16

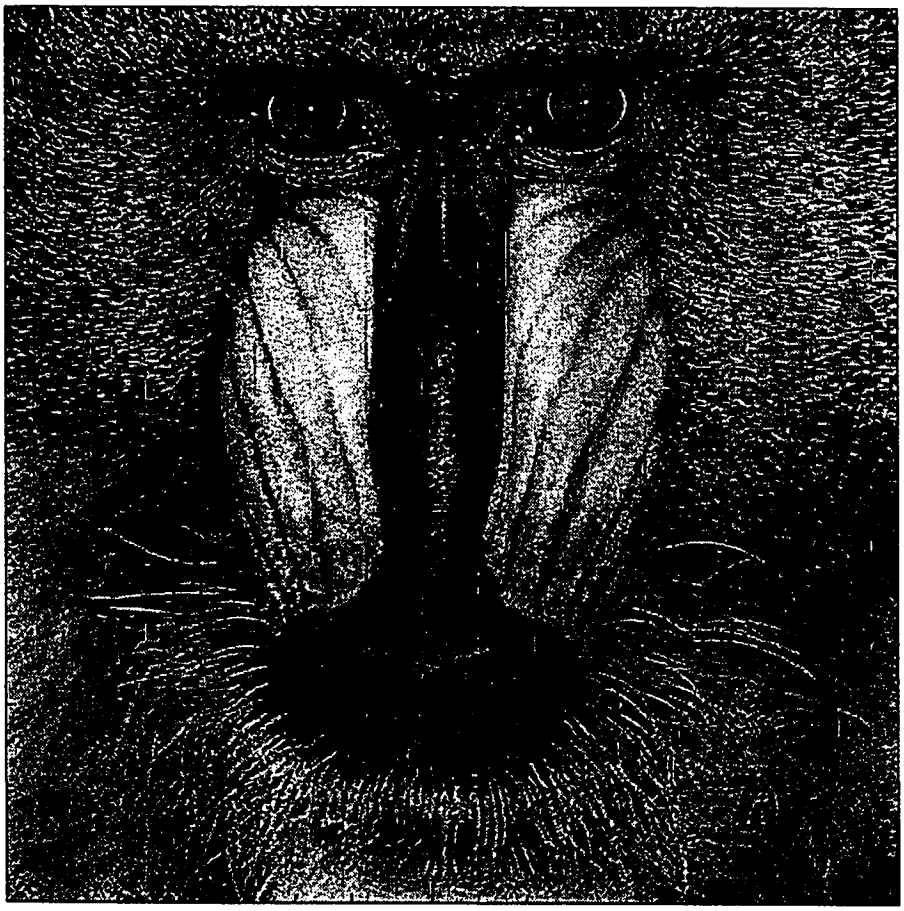


Figure 4.20 : SBMR image for block size 16 X 16 .

SHRI. RMSE=0.0800, SNR=7.7288, Ratio=4, Block size=16X16

Figure 4.21 : SHMR image for block size 16 X 16 .

NSBRI. RMSE=0.0836, SNR=7.6622, Ratio=4, Block size=16X16

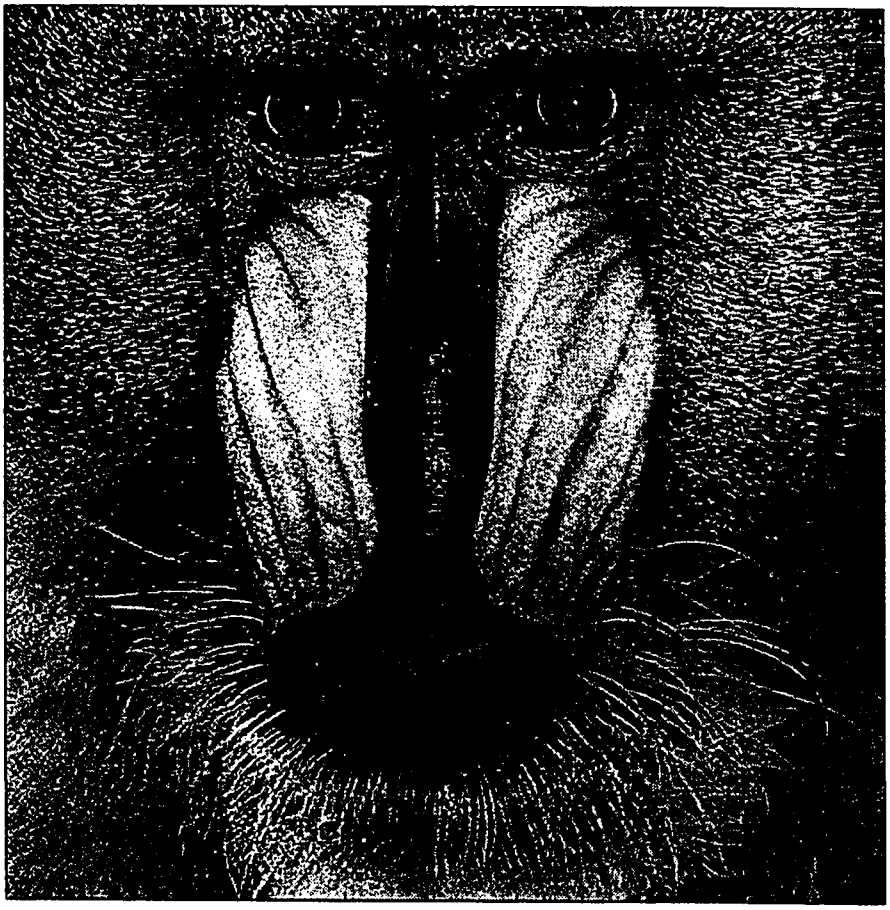


Figure 4.22 : NSBMR image for block size 16 X 16 .

NSHRI. RMSE=0.0781, SNR=7.9637, Ratio=4, Block size=16X16

Figure 4.23 : NSHMR image for block size 16 X 16 .

DCTI. RMSE=0.0971, SNR=5.2427, Ratio=4, Block size=16X16

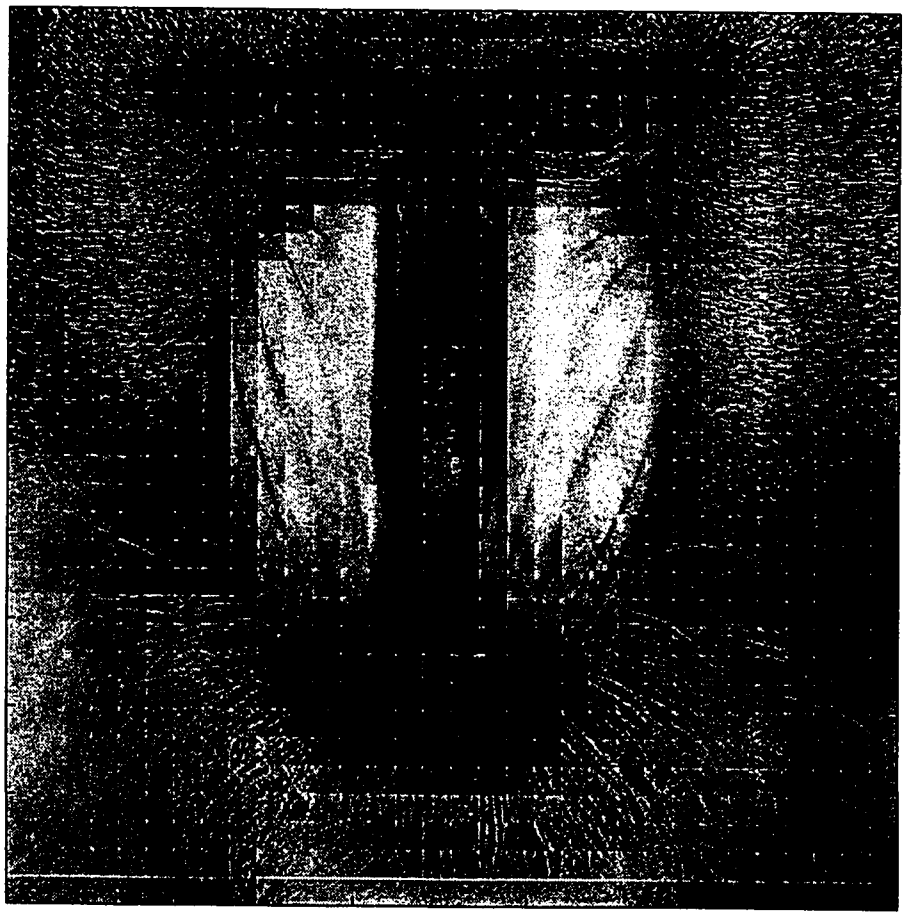


Figure 4.24 : DCT image for block size 16 X 16 .

SBRI. RMSE=0.1142, SNR=4.9763, Ratio=8, Block size=16X16

Figure 4.25 : SBMR image for block size 16 X 16 .

SHRI. RMSE=0.1099, SNR=4.8919, Ratio=8, Block size=16X16

Figure 4.26 : SHMR image for block size 16 X 16 .

NSBRI. RMSE=0.1128, SNR=5.0918, Ratio=8, Block size=16X16

Figure 4.27 : NSBMR image for block size 16 X 16 .

NSHRI. RMSE=0.1082, SNR=5.0886, Ratio=8, Block size=16X16

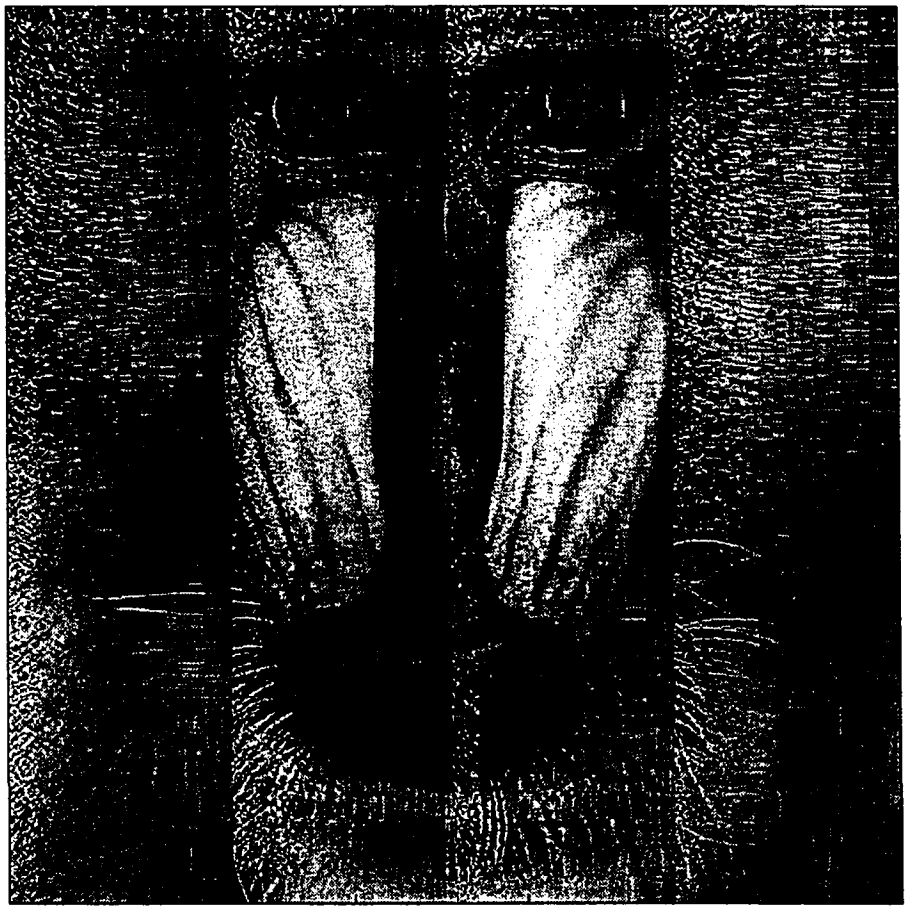


Figure 4.28 : NSHMR image for block size 16 X 16 .

DCTI. RMSE=0.1008, SNR=4.8146, Ratio=8, Block size=16X16

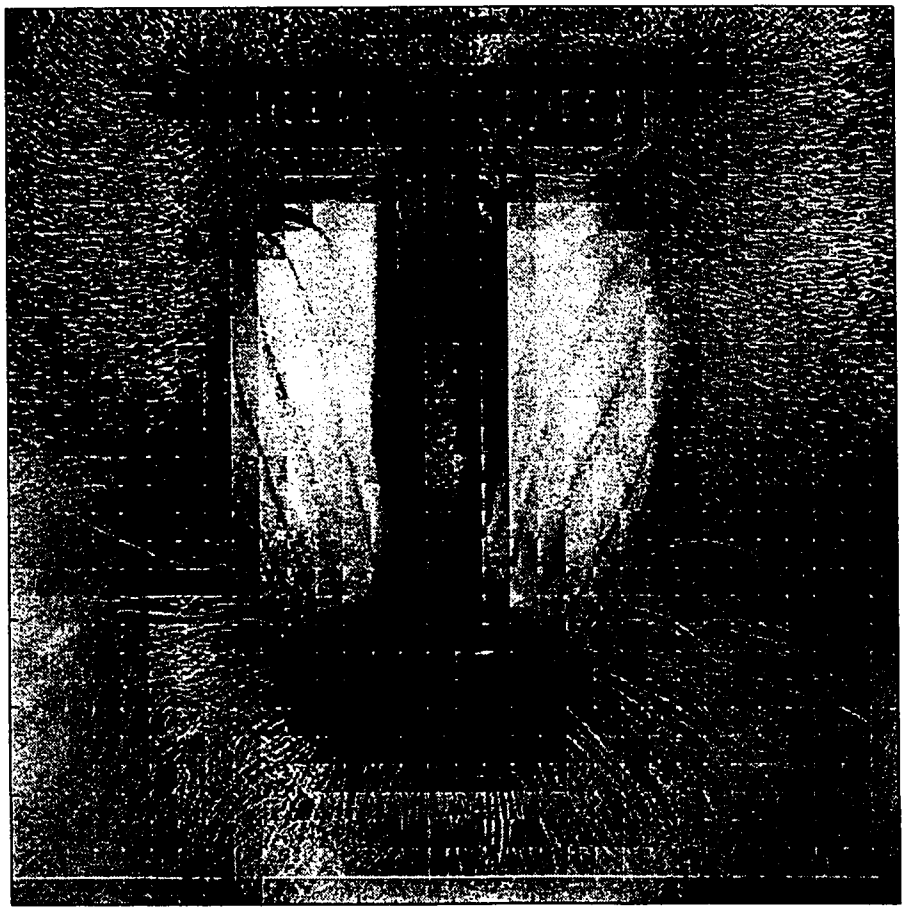


Figure 4.29 : DCT image for block size 16 X 16 .

The Signal-to-Noise Ratio(SNR) of the Hankel approximation schemes are also the largest followed by the balanced then the DCT scheme. Higher SNR indicates greater possibility of transmission of the approximated images with error-free communication.

The Hankel approximated images are also closer to the original as indicated by their correlation coefficients and the subjective parameters. The Hankel and Balanced approximated images also possess better contrast over the DCT approximated images. In addition the DCT approximated image suffers from the 'blocking artifact' phenomenon which is not so for the other approaches.

In terms of energy compaction the DCT scheme surpasses the other schemes. The DCT reconstructed images appears increasingly blurred as fewer number of coefficients are progressively retained (i.e., as the compression ratio increases) whilst for the balanced and Hankel compression schemes the reconstructed images only get degraded within some region.

4.4.3 Case III

In this example, Algorithm 6 primarily is implemented to analyse its performance. Comparison is also made to Algorithm 5. One of the tests images used is the image 'Mandrill'. This image is also processed at 8 bits/pixel amounts implying 256 gray levels. The image is concatenated row-wise(step 1 of algo-

rithm 6) and sampled in frames of length 128 (step 2). The elements of the frame are then grouped in twos (i.e. $g = 2$). Their respective spectral norms are then computed and the subsequent steps of Algorithm 6 are then applied repeatedly to each of subsequent frames .

In step 11 of the algorithm , the criterion for terminating the execution of the algorithm was 2 iterations. This was found to be sufficient based on simulations carried out with the number of iterations varying from 10 down to 2. There is no difference in terms of image quality with either two or ten iterations. The higher the number of iterations, the higher computational cost. Thus, the number of iterations was fixed at 2 for all runs.

The choice of the group size 'g' is taken as 2 (step 3 of the Algorithm), this was influenced by our numerous simulations carried out using other group sizes like 4, 8, 16 and others which showed that a group size of 2 was adequate enough. We found that the blocking effect on the reconstructed image tends to get more pronounced as the group size increases. This is thus a limiting factor on how much compromise can be accomodated in the choice of 'g' .

Tests were also conducted using other frame sizes; typical values were 64 and 32. We found that although the reconstructed image remains intelligible with decresing frame sizes, the computational time increases. We have adopted the training sequence (step 7) to be within four to five times that of the frame

length, this was influenced by computational considerations as the processing time increases with increasing length of the training sequence and the fact that we found this choice to be suitable.

In essence, as a *rule of thumb*, we have adopted the following as our standard parameters :

$$g = 2$$

$$iterations = 2$$

$$training\ sequence\ length = 4 - to - 5\ times\ frame\ length$$

Although approximations for different orders were carried out, only a representative sample of these approximations are illustrated here. The first-order approximation is mainly illustrated as its fidelity will easily give a measure of the performance of the algorithm since the first order image approximants may be considered as the 'worst case'. It is obvious that with increasing order the image quality will improve.

Figures 4.30 through 4.35 shows H_∞ norm different orders approximated images for certain specified values of parameters i.e- frame size and grouping for the image Mandrill whilst Figure 4.36 shows another original image type with its first order approximate in Figure 4.37.

It can be seen from Table 4.19 that in terms of reconstruction errors, the H_∞ norm approximated image compared to an equivalent DCT compressed case

s–approx im frame–128,order–1,win–2

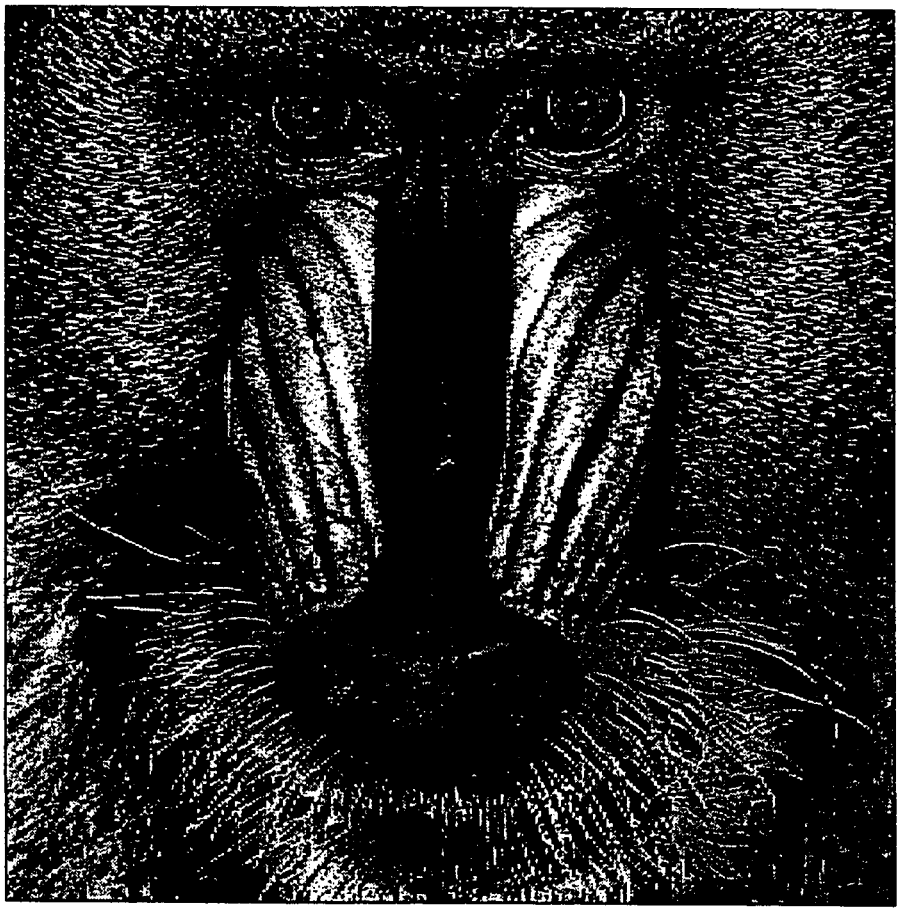


Figure 4.30 : Sorted approximated image with R=60.15% .

s–approx im ord–5,trseq–600,iter–2,fr–128,win–2

Figure 4.31 : Sorted approximated image with R=53.91% .

s–approx im ord–5,trseq–600,iter–2,fr–64,win–4

Figure 4.32 : Sorted approximated image with R=79.68 .

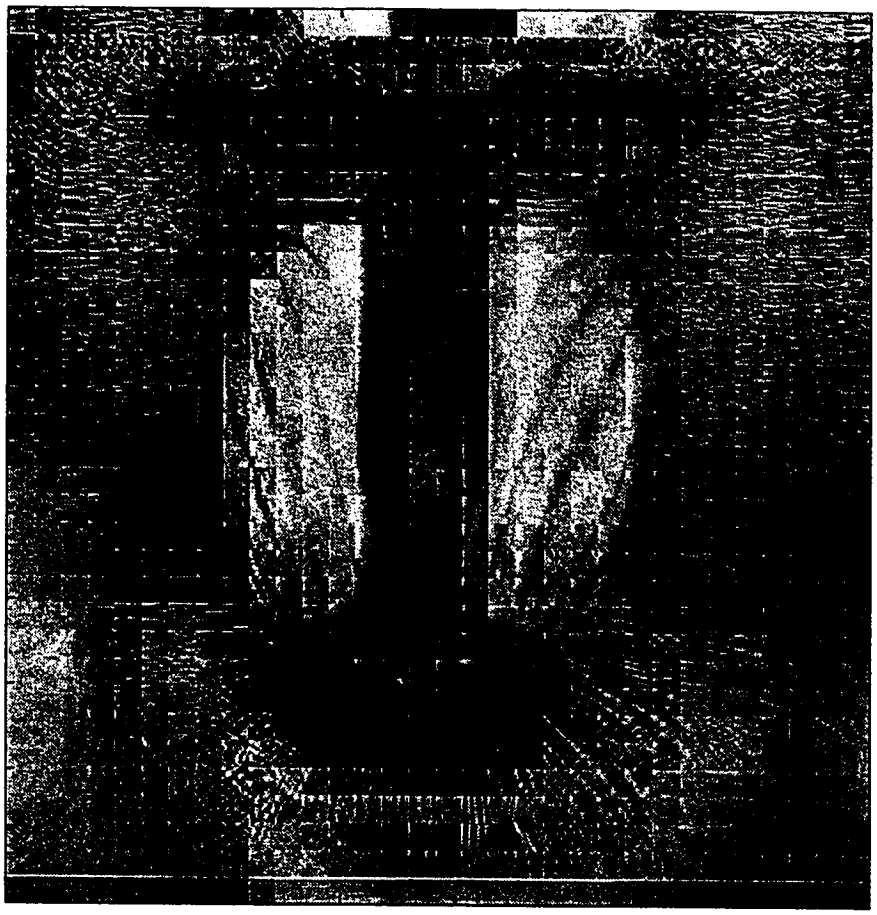


Figure 4.33 : DCT image with R=50.00% .

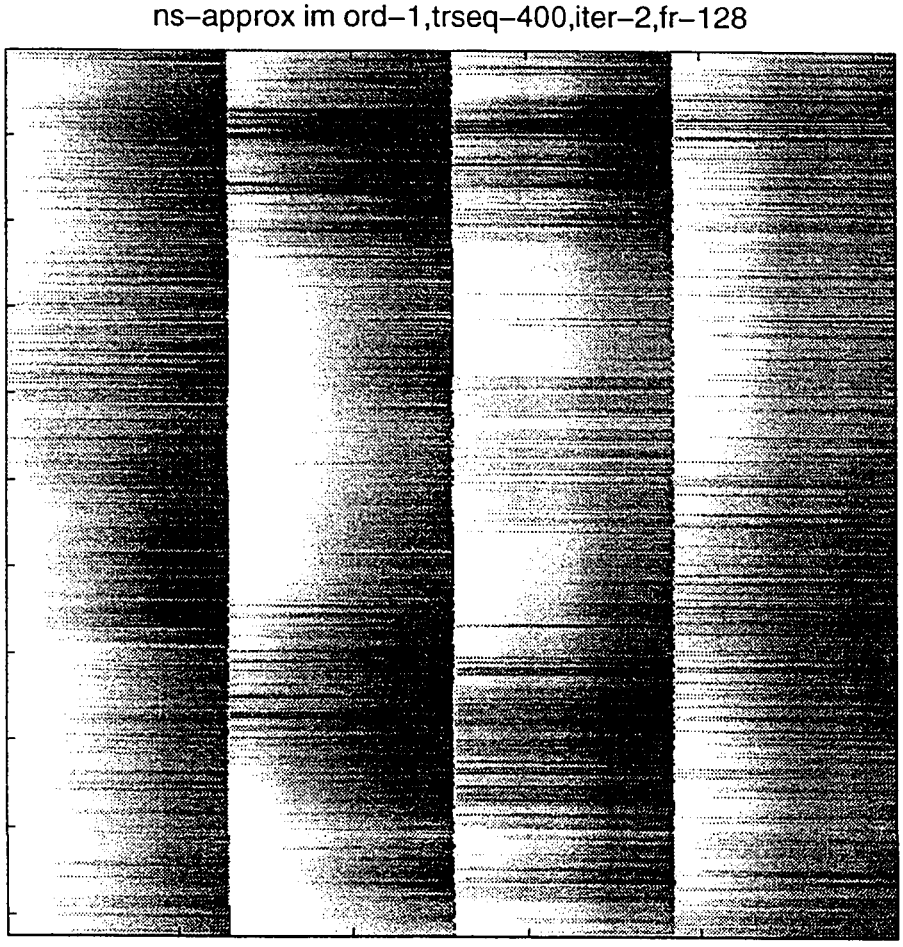


Figure 4.34 : Non-sorted approx. first order image.

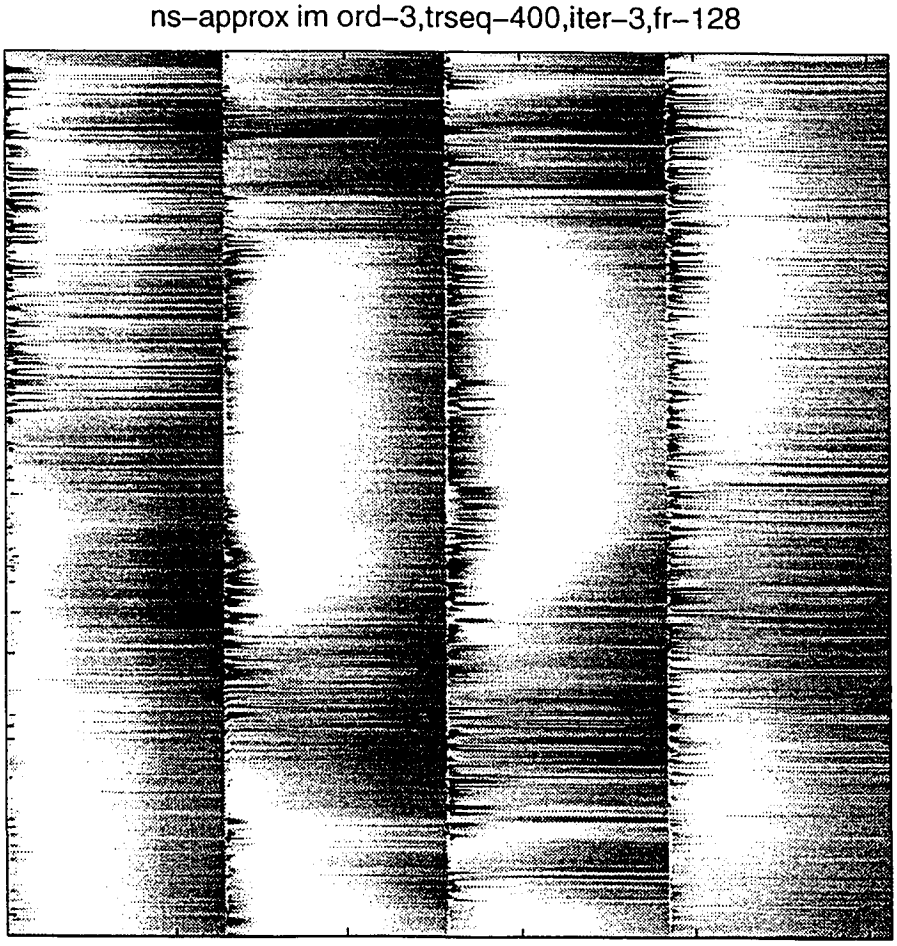


Figure 4.35 : Non-sorted approx. third order image.

Figure 4.36 : Original 'Bridge Scenery' image .

Figure 4.37 : First order approx . of 'Bridge Scenery' image .

has less errror.

The H_∞ norm approximated image also preserves fine details of the image (e.g. the sharpness of the eye of the image Mandrill) compared to the DCT compressed image which progressively gets blurrer with increasing compression ratio. In addition from Figure 4.30 we can see that the edge preservation is okay while in Figure 4.37the edge preservation is fair.

Our closed form expression for computing the reduction ratio (in percentage)in this case is given thus:

Without subgrouping

$$R = \left(1 - \frac{L' \times b + I \times b_i}{L \times b}\right) \times 100\% \tag{4.11}$$

and With subgrouping

$$R = \left(1 - \frac{L' \times b + B \times b_{ib}}{L \times b}\right) \times 100\% \tag{4.12}$$

where

L : frame length(number of original coefficients)

L' : approximated length(number of coefficients in approximant)

b : bits/pixel

I : length of indices of sorted coefficients

l : subgroup size(length)

B : number of subgroups = $\frac{L}{l}$

b_i : minimum number of bits required for index of sorted coefficients

b_{ib} : minimum number of bits required for subgroup index

n : order of approximation

4.4.4 Case IV

In this example, the performance of algorithm 7 is analysed. The speech signal used is an 8-bit mu-law format, of content 'Bismillahi Rrahmani Raheem'. The total duration of the speech signal is 4473.29 milliseconds. When sampled at 8012Hz, this speech signal produces 35840 samples requiring 35Kbytes of memory. Its frequency response plot is shown in Figure 4.38.

A brief review of the speech signal analogue of the results in the last section will be given here. Although the analysis of the speech signal is similar to that of the image signal, there are certain differences between the two.

Here we are only dealing with a 1-D signal/sequence as opposed to the 2-D case in the previous section. In addition the analysis frame (crossreference section 3.7) in this case is limited to sizes of either 8 or 16. Simulations were carried out using higher frame sizes i.e 32,64,128 and resulted in a reproduction of the speech signal which, although intelligible, was annoying to the ear. Since we wish to have a short duration window (impulse response) to be responsive

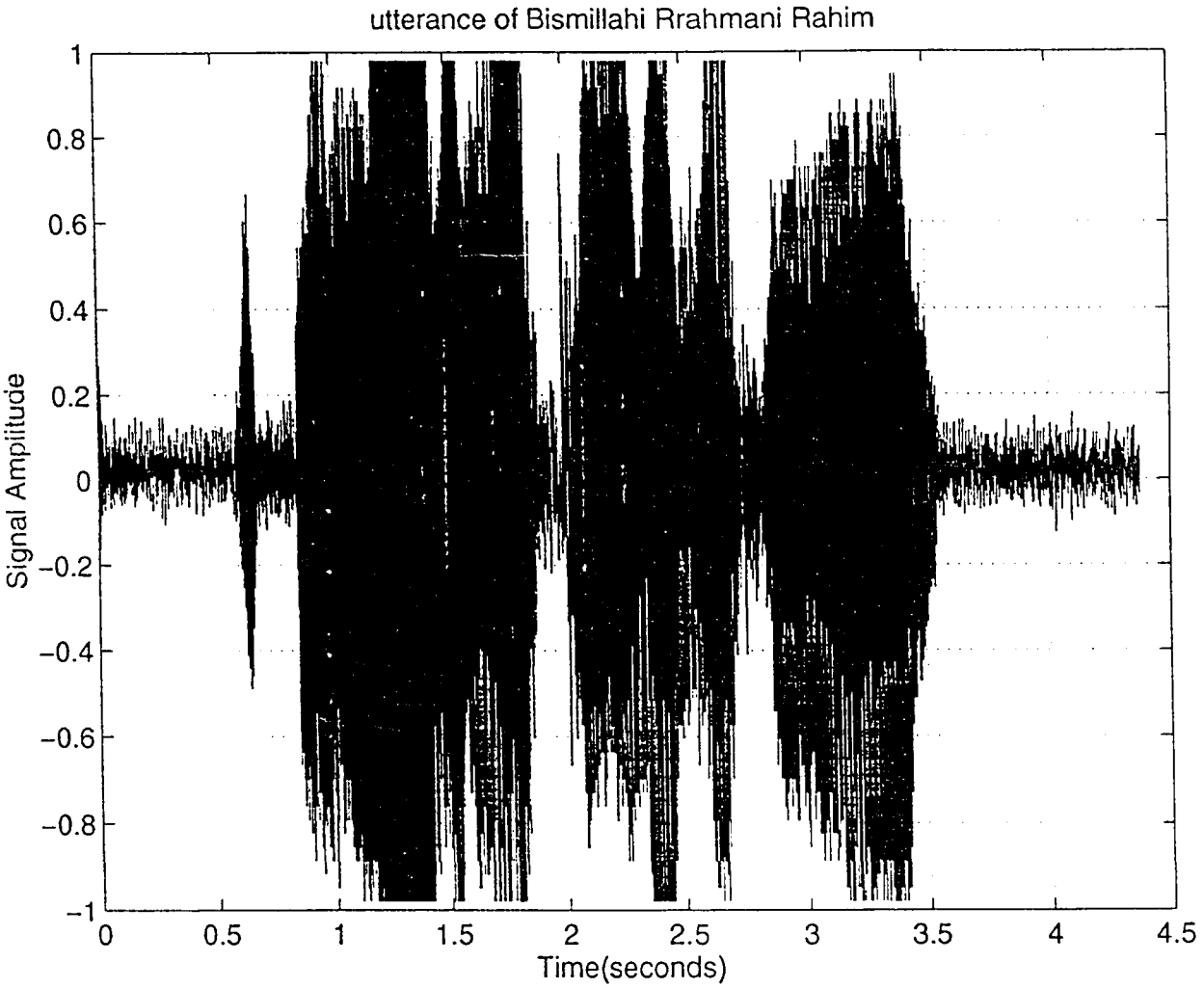


Figure 4.38 : 'Bismillahi R R' speech signal .

to rapid amplitude changes, we adopt frame sizes of 8 and 16 to evaluate the performance of the algorithm. It should be cautioned that a window that is too short will not provide sufficient averaging to produce a smooth energy representation within the frame! A too-long window might also violate the stationary assumption made for each analysis frame!

As in the case of the image signal; the group size, training sequence length, number of iterations, are all unchanged. The rest of the analysis required to obtain the approximated speech model is completely analogous to that carried out in the case of the image signal, it will therefore not be repeated here. The compression(reduction) formulas(equations 4.11 and 4.12) of the previous section is also valid for here.

Figures 4.39 and 4.40 shows the speech amplitude plots for different compression ratios; 50% and 75% respectively.

The approximated speech signals obtained were such that both the intelligibility and naturalness of the original was retained to a moderate degree.

4.5 Summary of Results

It was demonstrated in sub-section 4.4.1 that based on the quantitative image fidelity criteria, the Hankel approximation scheme performed better than the Balanced scheme in compressing images while retaining an adequate degree of

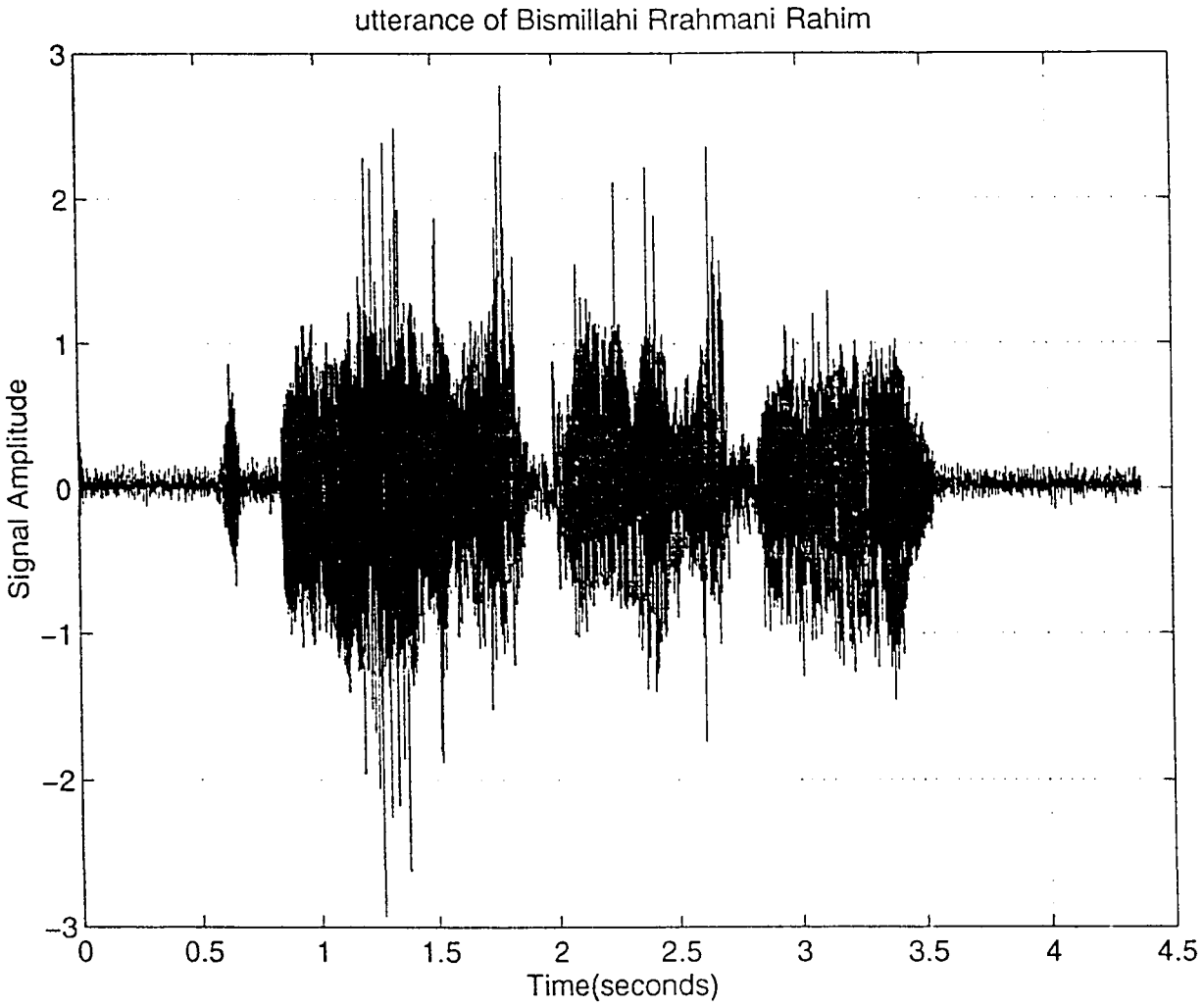


Figure 4.39 : First order speech signal for R=50% .

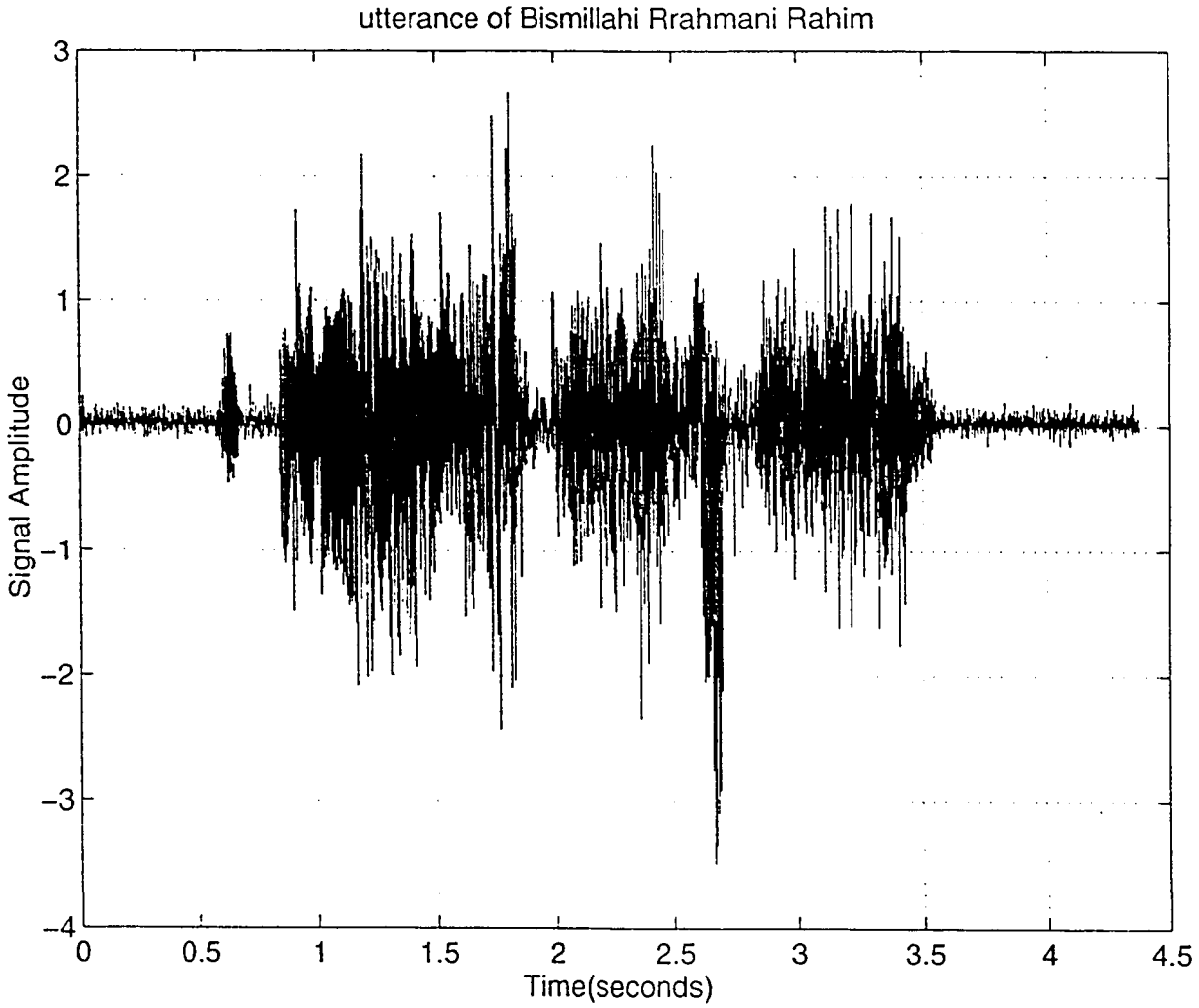


Figure 4.40 : First order speech signal for R=75%.

intelligibility in them.

From sub-section 4.4.2 we can see that the performance of the Hankel approximation schemes is superior to the Balanced and DCT schemes as indicated by their quantitative and qualitative image fidelity criteria.

In addition the overall performance of the H_∞ norm system approximated image is better than Hankel, Balanced and DCT schemes compressed image(s) as deduced from sub-sections 4.4.1, 4.4.2 aand 4.4.3. However, the DCT scheme surpasses them all only in terms of energy compaction.

Moreover, the H_∞ norm system approximation, Hankel and Balanced schemes do not suffer from blocking artifacts as can be seen from sub-sections 4.4.1, 4.4.2, and 4.4.3, thus the need for post-compression processing to remove this effect does not arise as is the case with the DCT scheme.

The edge reproduction of the decompressed-compressed images produced by the H_∞ norm algorithm are good as seen in the various examples(e.g. Figures 4.30, 4.31, 4.37 e.t.c.) in which fine details pertaining to edges/points remain undistorted.

It is worth mentioning that the non-sorted versions implementations of Algorithms 6 and 7 yielded very poor results in terms of intelligible reproduction of the decompressed-compressed signal as illustrated by two given examples (i.e. Figures 4.34 and 4.35) for different orders. It can be seen that this poss-

eses virtually no information at all. Some approaches (for instance negating the original image before applying the algorithms and reversing the negation process on the compressed image) were pursued to see if this problem could be rectified since this will amount to a very high compression ratio, however these approaches did not yield any encouraging results and were therefore abandoned. Thence, their non-sorted versions of were not pursued any further.

Developing a simple model capable of synthesizing intelligible images is not an easy problem. In this work, the algorithms that make use of state-space realizations (Algorithms 1-4) incorporates a relatively simple state-space image model compared to the traditional two-dimensional state space models. Moreover the need for separability of the transfer function denominator does not arise and thence does not limit the application of these algrotihms to only a certain class of 2-D systems(i.e. separable) as is the case with existing methods of 2-D model reduction. On the other hand, algorithms 6 & 7 take care of the cumbersome problem of state-space realization, when the system order is large, via identification techniques.

Since the algorithm samples the original image in blocks of smaller size, the complexity of the hardware for implementing small-size processing systems is siginficantly reduced, hence offering the possibility of parallel processing.

The main limitations in implementing compression algorithms are memory and computational time. Precisely, note that a practical consideration in the implementation of compression algorithms is the sub-image size. The choice depends on available memory size. In general the level of compression and computational complexity increases as the sub-image block size increases.

Thus, given an overall objective, the constraints imposed and information about the class of images to be processed, a reasonable approach to developing an image compression scheme(algorithm) can be determined.

Chapter 5

Conclusions and Recommendations

5.1 Conclusions

The studies performed in this thesis primarily involved developing and implementing compression algorithms for image data compression. The algorithms proposed in this work are for still-image data compression. The noise-free case has be assumed in the application of the algorithms. The developed and implemented algorithms with the exception of the DCT-based method are all new techniques for image data compression. One of the algorithms (i.e. Algorithm 6) is applied to a speech signal.

The developed algorithms were implemented through on-line simulation studies for different image types and a particular speech signal. The overall performance of these algorithms based on extensive simulation studies are quite encouraging. Thus these new techniques posses great potential in them as new and acceptable standard approaches for image data compression/coding/ or approximation. They may also be extended to speech compression/coding/ or approximation .

It is concluded that these methods are promising and worth pursuing further.

5.2 Recommendations for Future Studies

During the course of this research, several interesting areas which could be pursued for the developed algorithms to evolve into standard method(s) of compression were identified. A good collection of these research areas are summarized below.

- Optimizing the algorithms for faster implementations.
- Further work in the area of speech compression i.e. extension to speech compression.
- Studying how even higher compression ratio can be achieved through a tradeoff in the number of bits required for storing/transmitting the

sorting indices, in the sorted cases.

Bibliography

[1] R C. Gonzalez and P E. Woods. *Digital Image Processing.* Addison-Wesley Pub. Co., 1992.

[2] L R. Rabiner and R W. Schafer. *Digital Processing of Speech Signals.* Prentice hall Inc, 1978.

[3] B. Furht. Multimedia systems: An overwiew. *IEEE Multimedia*, pages 47–59, 1994.

[4] A K. Jain. *Fundamentals of Digital Image Processing.* Prentice-Hall Inc, 1989.

[5] J S. Lim. *Two Dimension Signal and Image Processing.* Prentice-Hall, 1993.

[6] W.S. Lu, E.B. Lee, and Q.T. Zhang. Model reduction for two dimensional systems. *Proc. IEEE Int. Symp. Circ. Systems*, 1:79–82, 1986.

[7] K. Premaratne, E. I. Jury, and M. Mansour. An algorithm for model reduction of two dimensional discrete time systems. *IEEE Trans. Circuits*

123

Syst., 37:1116–1132, Sept. 1990.

[8] N T. Theodorou and S G. Tzafestas. Reducibility and factorizabilty of multivariable polynomials: Overview and new results. *Cont. Theory and Adv. Technology*, 1:25–46, Apr. 1985.

[9] G E. Antoniou, P. N. Paraskevopoulos, and S.J. Varoufakis. Minimal state-space realization of factorable two dimensional transfer functions. *IEEE Trans. circuits syst.*, 35:1055–1058, Aug. 1988.

[10] A. Kumar, F. W. Fairman, and J. R. Sveinsson. Separately balance realization and model reduction of two dimensional separable denominator transfer functions from input-output data. *IEEE Trans. on ccts and systems.*, CAS-34, no.3:233–239, Mar. 1987.

[11] B. Lashgari, L. M. Silverman, and J F. Abramatic. Approximation of two dimension separable in denominator filters. *IEEE Trans. Circuits syst.*, CAS-30, no. 2:107–121, Feb. 1983.

[12] S. J. Varoufakis, P. N. Parskevopoulos, and G. E. Antoniou. On minimal state-space realizations of all-pole and all-zero two dimensional systems. *IEEE Trans. Circuits syst.*, 34(3):289–292, Mar. 1987.

[13] K. Zhou, Y. Li, and E. B. Lee. Model reduction of two dimensional systems with frequency error bounds. *IEEE trans.circuits syst-II: Analog and Digital Procesing*, 40(2):107–110, Feb. 1993.

[14] T. Lin, M. Kawamata, and T. Higuchi. Design of two dimensional seperable denominator digital filters based on the reduced dimensional decomposition. *IEEE Trans. circuits syst.*, 34(8):934–941, Aug. 1987.

[15] T. Lin, M. Kawamata, and T. Higuchi. Decomposition of two dimensional seperable denominator systems: existence, uniqueness and applications. *IEEE Trans. circuits syst.*, 34(3):292–296, Mar. 1987.

[16] T. Lin, M. Kawamata, and T. Higuchi. On controllability, observability and minimality of two dimensional seperable denominator sytems: A new approach based on the reduced dimensional decomposition. *IEEE Trans. circuits syst.*, 34(8):962–967, Aug. 1987.

[17] P. Misra and T. Manickam. Balanced realization of seperable denominator multidimensional systems. *Lin. Alg. Appl.*, pages 521–547, 1993.

[18] R P. Roesser. A discrete state-space model for linear image processing. *IEEE Trans. Auto Contr.*, pages AC20: 1–10, 1975.

[19] E. Fornasini and G. Marchesini. State-space realization theory of two-dimensional filters. *IEEE Trans. Auto. Contr.*, pages AC–21:484–492, 1976.

[20] R. Eising. Realization and stabilzation of two-dimensional systems. *IEEE Trans. Auto. Contr.*, pages 793–799, Oct. 1978.

[21] E. Fornasini and G. Marchesini. A critical review of recent results on two dimensional system theory. *IFAC control science and technology (sth triennal world congress) kyoto, Japan*, pages 255–269, 1981.

[22] S.Atassi. Systemes lineaires homogenes a deux indices. *Rapport LABORIA, 31*, 1973.

[23] R M. Mersereau and D E. Dudgeon. The representation of two dimensional sequences as one dimensional sequences. *IEEE Trans. on Acoustics, Speech and Signal Processing*, pages 256–260, Oct. 1974.

[24] M T. Manry and J K. Aggarwal. Picture processing using one dimensional implementations of discrete planar filters. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, pages 164–173, Jun. 1974.

[25] J C. Doyle, B A. Francis, and A R. Tannebaum. *Feedback Control Theory.* Macmillan Inc, 1992.

[26] G.W. Stewart. *Introduction to Matrix Computations.* Academic Press, 1973.

[27] B C. Moore. Principal components analysis in linear systems: Controllability, observability and model reduction. *IEEE Trans. on Auto. Contr.*, AC-26:17–32, 1981.

[28] M. Bettayeb. *Approximation of Linear Systems: New Approaches based on Singular Value Decomposition.* PhD Thesis, USC, 1981.

[29] K. Glover. All optimal hankel norm approximations of linear multivariable systems with their l_∞ error bounds. *Int. J. Control*, 39(6):1115–1193, 1984.

[30] T Soderstorm and P Stoica. *System Identification.* Prentice Hall International, 1989.

[31] D. Kavranoglu and M. Bettayeb. Characterization of the solution to the h_∞ model reduction problem. *Systems and Control Letters*, 20:99–107, 1993.

[32] D. Kavranoglu and M.S. Ahmed. h_∞ norm system approximation by system identification technique. *submitted to IEEE Trans. Auto. Contr.*, Dec. 1994.

[33] D. Kavranoglu. *Elementary solutions for the H_∞ General problem- Equivalence of H_2 and H_∞.* PhD Thesis, Caltech, 1989.

[34] W B. Pennebacker and J L. Mitchell. *JPEG Still Image Data Compression Standard.* Van Nostrand Reinhold, New York, 1993.

[35] C.T. Chen. *Linear Systems Theory and Design.* Rine Hart Pub. Co, 1994.

[36] S. Furui and M.M. Sandhi (eds). *Advances in Speech Signal Processing.* Marcel Dekker Inc, 1991.

[37] J.D. Markel and H.H. Gray. *Linear Prediction of Speech.* Springer-Verlag, 1976.

[38] W.B. Kleijn, D.J. Kransinski, and R.H. Ketchum. Fast methods for the celp speech coding algorithms. *IEEE Trans. on Acoustics, Speech and Signal Processing*, ASSP-38(8):1330–1341, 1990.

[39] S R. Quackenbush, T P. Barnwell III, and M A. Clements. *Objective Measures of Speech Quality.* Prentice Hall Inc, 1988.

[40] G A. Miller. The magic number seven plus or minus two. *In S. Ducker(Ed), Time-Compressed Speech (Vol.1) Methchen, N.J. The Scarecrow Press*, 1974.

Vita

- Jibril Odogba
- Received Bachelor's degree in Electrical Engineering from Ahmadu Bello University, Zaria, Nigeria in June 1989.
- Completed Master's degree in Systems Engineering at King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia in June, 1995.